Выявление пересекающихся сообществ в социальных сетях

Бузун Назар* Коршунов Антон†

20 февраля 2012 г.

*nazar@ispras.ru, Институт системного программирования РАН, Россия, г. Москва

†korshunov@ispras.ru, Институт системного программирования РАН, Россия, г. Москва

Аннотация

Кластерная структура является одной из главных особенностей социальных графов. Несмотря на большое количество алгоритмов ее выявления, существует необходимость определения области их эффективной применимости при различных значениях конфигурационных параметров сети. В этой статье основное внимание уделено степени пересечения кластеров. Выполнено тестирование как наиболее современных методов нечеткой кластеризации, так и обобщенных классических подходов. В зависимости от величины пересечения сделан вывод о применимости отдельных классов алгоритмов с общей методикой и их представителей.

Ключевые слова: Кластеризация, социальные сети, выявление сообществ, community detection

Введение

Сети являются естественным представлением различных сложных систем в обществе, биологии, технике и других областях. Множество сетей характеризуются мезоскопическим уровнем организации внутри групп узлов, образующих единицы с большим количеством связей. Такие единицы называются **кластерами** (**сообществами** или **модулями**).

В последние годы внимание данной области исследований сфокусировано на социальных и естественных сетях, для

обнаружения внутренней структуры которых не применимы классические алгоритмы кластеризации.

Можно привести несколько вариантов полезной информации, полученной на основании разбиения сети на сообщества: обнаружение функциональных единиц системы, выявление сходства между вершинами из одного сообщества, вершины в одном сообществе могут классифицироваться в соответствии с их позицией (лидеры, связывающие и т.д.), удобный способ визуализации системы, определение атрибутов вершин на основании общих атрибутов сообществ, включающих данные вершины [1].

Рассмотрим некоторые особенности структуры социальных сетей, которые требуется учитывать при выборе алгоритма кластеризации:

1. Вершина может находиться одновременно в нескольких сообществах с разной степенью принадлежности (**fuzzy clusters**) [2-10,24]

2. Сообщества могут иметь **иерархическую структуру** [4,6,8,11,22,24], что требуется для эффективного управления в масштабных организациях, а наличие таковой подчеркивает стабильность системы [12]

3. Помимо того, высокая плотность ребер не всегда свидетельствует о наличии кластера. Поэтому для отсечения "псевдосообществ" вычисляется вероятность реализации конкретной конфигурации подграфа ("**статистическая значимость**") в предположении истинности гипотезы случайного распределения ребер (при заданных значениях степеней вершин) [5,9,13].

4. В некоторых случаях (например при определении ат-

рибутов вершин) необходимо присвоить вершинам и ребрам **несколько параметров** [1,2,14]. При этом в большинстве своем в настоящее время алгоритмы принимают на вход лишь 1 параметр - веса ребер.

5. Также может дополнительно ставиться задача изучения **динамики сообществ** в сети [15].

В данной статье внимание акцентируется на выявлении пересекающихся сообществ в сетях больших размеров (n $=10^8, m \sim n$) с высоким коэффициентом пересечения ($r \sim 10$). Где P-множество сообществ $G = (E, V), m = |E|, n = |V|, r = \sum |P_i|/n$. Приводиться несколько современных алгоритмов, которым изначально присуща возможность выявления указанных сообществ. Помимо того, предлагается несколько вариантов обобщения классических алгоритмов на случай графов с пересекающимися кластерами. Целью же данного исследования является выявление наиболее релевантных методов нечеткой (пересекающейся) кластеризации и способов оценки качества разбиения графа.

Обзор методов выявления сообществ

1 Модель случайного графа (null model)

В методах данного класса заданная конфигурация ребер сравнивается с равномерным их распределением для каждой вершины графа. При этом степени вершин случайного графа в большинстве случаев считаются известными параметрами. Классическим вариантом здесь является максимизация це-

левой функции **modularity** и ее модификаций [16-21], характеризующей суммарную разность количества ребер в сообществе и их математического ожидания:

$$Q = \frac{1}{2m} \sum_{c \in Pi, j \in c} [A_{ij} - Pr(A_{ij} = 1)]$$

P - множество сообществ, A-матрица инцидентности.

Аналогично вместо ребер можно брать во внимание количество треугольников в сообществе, считая связи между вершинами слабыми, если они не являются ребрами треугольника [8,20].

Изначально modularity вводилась для характеристики непересекающихся разбиений, но существуют ее обобщения и на случай **пересекающихся сообществ** [17,21]. Помимо того, стоит упомянуть ее квантовомеханическую модификацию [18,19], позволяющую улучшить **разрешающий предел** и придающую ей энергетический смысл системы вершин с различными спиновыми значениями (**spinglass** [19]).

Более общим методом является обнаружение "значимых" кластеров, имеющих малую вероятность конфигурации в предположении истинности гипотезы случайного графа. (**oslom, moses**) [5,9]

2 Блуждания (random walk)

infomap[10,22]: В данном случае граф разбивается таким образом, чтобы минимизировать длину описания случайного блуждания в данном графе. Одной из оценочных функций

для ожидаемой длины кода является энтропия, широко используемая в различных разделах теории информации. Исходя из этого в [22] предлагается в качестве оценочной рассматривать следующую функцию:

$$L(P) = qH(Q) + \sum_i p_i H(P_i)$$

где q-вероятность перехода в другой модуль, p_i- доля переходов внутри i-го модуля, H(Q)-энтропия названий модулей, H(P_i)-энтропия названий вершин внутри модуля.

walktrap[23]: Здесь формирование сообществ происходит на основе следующего утверждения: Пусть вершины i , j принадлежат одному кластеру, тогда Pr(k|i,t) $\approx$ Pr(k|j,t) для всех $k \in V$, где Pr - матрица перехода.

betweenness[9]: Введение меры "промежуточности" на множестве ребер (чем больше проходов по ребру при случайном блуждании, тем больше величина меры). Ребра с большой "промежуточностью" естественно считать внекластерными **(conga, GN)** [3].

3 Локальный анализ подграфов

При локальном изучении и формировании кластера (без учета структуры остальной части графа) обычно рассматривается отношение количества внутренних ребер и треугольников к внешнему и максимально возможному их числу (**cohesion** [8]) . Также сообщества могут формироваться на основании схожести с полным графом или набором связанных клик различного размера **(CFinder, GCE)** [7]. Помимо того, для

локальной характеристики похожести подграфа на сообщество может быть использована упомянутая выше **“статистическая значимость**”. Существует также набор методов из данного раздела, позволяющих независимо выделять подграфы с высокой величиной влияния вершин внутри себя (**moduland** [6]). Для данного класса методов свойственно естественное выделение пересекающихся сообществ, но возникают трудности с последующим формированием конечного разбиения всего графа.

4 Введение координат

Еще одним изящным подходом является присвоение координат вершинам в графе [26], которыми являются компоненты собственных векторов нормированной матрицы Лапласа **L**. Данный способ кластеризации является крайне полезным, если требуется использовать уже известные атрибуты вершин.

Подытоживая обзор, можно выделить методы spinglass, infomap, wolktrap обладающие наиболее высокими показателями **Normalized Mutual Information** [24] (для случая **непересекающихся** сообществ) при относительно небольшом времени работы и возможностью параллельного исполнения [25].

Способы обобщения на случай пересекающихся сообществ

1 Статическое

Используя меру betweenness на множестве вершин, можно каждую вершину с высоким значением меры разделить на две, соединенные ребром (**toolteep** [3]). Альтернативный вариант - генерация линейных графов(ребра переходят в вершины, а вершины в ноль или несколько ребер) и последующая кластеризация ребер [4].

2 Динамическое

Вводя коэффициенты принадлежности для вершин (вероятности нахождения в каждом из сообществ), в процессе работы алгоритма относят вершину одновременно к нескольким кластерам. Как ориентир для коэффициента принадлежности могут быть использованы следующие величины:

Индивидуальный вклад в прирост целевой функции:

$$Pr(V_i \in P_k) \sim Q(V_i \in P_k) - Q(P_k \backslash V_i) = \triangle Q_{ik}$$

Вероятность нахождения на определенном энергетическом уровне:

$$Pr(V_i \in P_k) = e^{-\beta Q(V_i \in P_k)} / \sum_S e^{-\beta Q(V_i \in P_S)}$$

Q - целевая функция, β-величина, обратно пропорциональная коэффициенту пересечения.

Интересно заметить, что введение коэффициентов принадлежности часто улучшает разбиение на непересекающиеся сообщества. Основная идея здесь в том, что задавая вероятностей переходов вершины в другие сообщества (оставаясь с определенной вероятностью в исходном) мы "сообщаем" другим вершинам тактику ее поведения. Т.о. для задания коэффициентов в этом случае может быть использовано следующее выражение:

$$Pr(V_i \in P_k) \sim \triangle Q_{ik} - \min_h(\triangle Q_{ih}),\ V_i \in P_h$$
$$Pr(V_i \in P_{hmax}) \sim 0.1$$

Способы реализации

1) Алгоритм жадной оптимизации целевой функции (используется большинством из упомянутых выше алгоритмов): Изначально каждая вершина является сообществом. Далее на каждом шаге каждая вершина выбирает к каким сообществам присоединиться, сравнивая величины прироста целевой функции. Завершающей стадией является объединение сформировавшихся модулей, с большим числом связей.

2) Метод центральных вершин [2]: Задается несколько центральных вершин, к которым постепенно присоединяются остальные, выбирая наиболее "близкий" кластер.

3) Рекурсивное разбиение исходного графа на две и более частей [16]. Вначале вершины разбиваются случайным образом. Затем перемещаются **в первую очередь** те, которые дают максимальный прирост целевой функции.

4) В случае применения локальной оптимизации предлагается следующая схема [9]:

Однокластерный анализ $\mapsto$ Проверка внутренней структуры $\mapsto$ Объединение кластеров $\mapsto$ Вычисление коэфф. принадлежности.

Тестирование

Здесь в качестве генератора сетей с пересекающейся кластерной структурой используется LFR benchmark алгоритм [9]. С целью исследования работы алгоритмов при различной величине пересечения сообществ были сгенерированы два множества тестовых графов с предопределенным разбиением. В качестве параметров указанному генерарору передавались следующие величины: n - число вершин, k - срернее значение степени вершины, kmax - максимальное значение степени вершины, $|P_i|$ - количество вершин в кластере, τ_1 - значение экспоненты степенного распределения степени вершин, τ_2 - значение экспоненты степенного распределения $|P_i|$, μ - усредненная нормированная степень вершины внутри родительского сообщества, on - число вершин, принадлежащих более чем одному сообществу, om - количество сообществ, содержащих фиксированную вершину. Параметры графов из первого множества отличаются значением om, из второго - значением on.

Для сравнения полученных разными методами разбиений (Рис1: fig1, fig2) будем использовать меру **Normalized Mutual Information (I_{norm})** [24], базирующуюся на следующем предложении: если два разбиения графа похожи, то

требуется относительно небольшое количество информации для получения первого разбиения при известном втором.

$$I(X,Y) = H(X) - H(X|Y)$$

$$I_{norm}(X,Y) = \frac{2I(X,Y)}{H(X)+H(Y)}$$

Где H - энтропия Шеннона.

В дополнение, для разбиений графов из первого множества (Рис1: fig3) будем вычислять обобщенную на случай нечеткой кластеризации функцию Modularity.

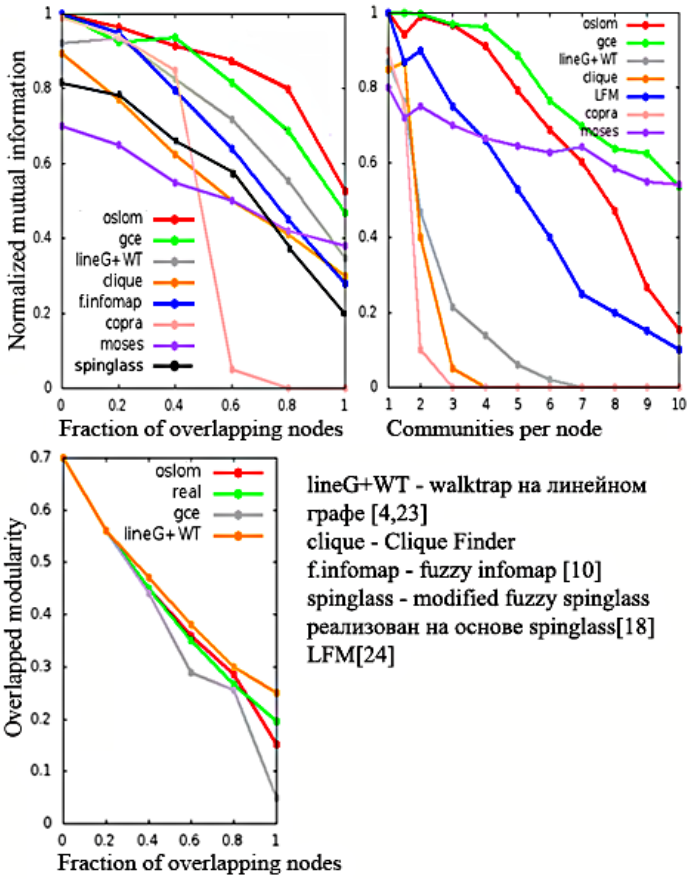

Рис.1: Тестирование алгоритмов нечеткой кластеризации.

fig1,fig3: $n = 2000, k = 15om, kmax = 45om, |P_i| \in [15, 60], \tau_1 = 2, \tau_2 = 0, \mu = 0.2, on \in \{0, 1000, 2000\}$, $om \in \{1, 1.5, 2, 3, ..., 9, 10\}$

fig2: $n = 1000, k = 20, kmax = 50, |P_i| \in [20, 100], \tau_1 = 2, \tau_2 = 1, \mu = 0.3, on \in \{0, 200, 400, ..., 1000\}$, $om = 2$

По результатам экспериментов выявления сообществ с разной величиной пересечения можно заключить, что для случая **значительного пересечения** может быть эффективно применено лишь несколько методов - **oslom** [9], **moses** [5], **gce** [7], которые представляют класс локальной оптимизации. В частности первые два свидетельствуют об эффективности использования "статистической значимости" в качестве индивидуальной(локальной) характеристики выраженной кластерной структуры. Также стоит отметить результативность подхода пересекающейся кластеризации ребер [4], который может быть применен для сетей средних и малых размеров. Для больших сетей с незначительным пересечением могут быть использованы методы, имеющие сложность не более $O(n^\alpha)$ $\alpha \in [1, 2]$ - **fuzzy infomap** [10], **gce** [7], **fuzzy spinglass** [18].

Помимо того, анализируя графики значений Modularity (fig3), следует подчеркнуть расхождение в оценке качества разбиения с I_{norm} при увеличении on. Откуда следует, что Modularity дает объективную оценку разбиения только при небольших значениях коэффициента пересечения r.

Заключение

В итоге, в данном исследовании было указано несколько основных свойств социальных и естественных графов, проведено разбиение алгоритмов на четыре класса. Также предложено несколько различных типов их обобщения на случай пересекающихся сообществ и приведены основные варианты их реализации. Из результатов тестирования на искусственно сгенерированных сетях выявлена применимость наиболее современных методов при различных конфигурациях графа.

Возможными направлениями дальнейшей работы является продолжение изучения слабых и сильных сторон приведенных классов алгоритмов в зависимости от свойств графа и поставленных целей. При этом во внимание будут приниматься все отмеченные в начале статьи особенности социальных графов. В частности, довольно значимыми являются задача выявления иерархической структуры, методов ее оценки, а также кластеризация графов с атрибутами (**ordered graphs** [14]) на множестве вершин и ребер - что является первостепенной задачей для предсказания неизвестных атрибутов. Вследствие аккумуляцией результатов проведенных исследований может быть обучающийся анализатор графов, определяющий на каких частях графа (отличающихся, например, величиной пересечения сообществ) может быть эффективно применен конкретный метод.

Список литературы

[1] Lei Tang. 2010. Learning with Large-Scale Social Media Networks. Ph.D. Dissertation. Arizona State University, Tempe, AZ, USA. Advisor(s) Huan Liu. AAI3425805

[2] Zhang S, Wang RS, Zhang XS. 2007. Identification of overlapping community structure in complex networks using fuzzy c-means clustering. Physica A 374: 483–490.

[3] Gregory S. 2007. An algorithm to find overlapping community structure in networks. Berlin, Germany: Springer-Verlag. pp 91–102. `https://www.cs.bris.ac.uk/~steve`

[4] Y Ahn, JP Bagrow, S Lehmann. 2010. Link communities reveal multi-scale complexity in networks. Nature 466, 761–764.

[5] AF McDaid, NJ Hurley. 2010. Using Model-based Overlapping Seed Expansion to detect highly overlapping community structure. In: ASONAM 2010. `http://sites.google.com/site/aaronmcdaid/moses`

[6] Kovacs IA, Palotai R, Szalay MS, Csermely P. 2010. Community landscapes: An integrative approach to determine overlapping network module hierarchy, identify key nodes and predict network dynamics. PLoS ONE 5: e12528

[7] Lee C, Reid F, McDaid A, Hurley N. 2010. Detecting highly overlapping community structure by greedy clique expansion. Poster at KDD 2010.

[8] A. Friggeri, G. Chelius, and E. Fleury. 2011. Egomunities, Exploring Socially Cohesive Person-based Communities. NRIA, Research Report RR-7535, 02 2011

[9] A. Lancichinetti, F. Radicchi, J. Ramasco, S. Fortunato. 2011. Finding Statistically Significant Communities in Networks. PLoS ONE 6(4): e18961. `http://santo.fortunato.googlepages.com/inthepress2`

[10] AV Esquivel, M Rosvall. 2011. Compression of flow can reveal overlapping modular organization in networks. Phys. Rev. X 1, 021025 (2011). `https://sites.google.com/site/alcidesve82`

[11] Clauset A, Moore C, Newman MEJ. 2008. Hierarchical structure and the prediction of missing links in networks. Nature 453: 98–101.

[12] Simon H. 1962. The architecture of complexity. Proc Am Phil Soc 106: 467–482.

[13] Lancichinetti A, Radicchi F, Ramasco JJ. 2010. Statistical significance of communities in networks. Phys Rev E 81: 046110

[14] Gregory S. 2011. Ordered community structure in networks. Physica A: Statistical Mechanics and its Applications (December 2011)

[15] Mucha PJ, Richardson T, Macon K, Porter MA, Onnela J. 2010. Community Structure in Time-Dependent, Multiscale, and Multiplex Networks. Science 328: 876.

[16] M. E. J. Newman. 2006. Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E 74, 036104 (2006)

[17] V. Nicosia,G. Mangioni,V. Carchiolo,M. Malgeri. 2008. Extending modularity definition for directed graphs with overlapping communities. J. Stat. Mech. P03024 (2009).

[18] J. Reichardt, S. Bornholdt. 2008. Statistical Mechanics of Community Detection. Phys. Rev. E 74 (1) (2006) 016110

[19] P. Ronhovde, Z. Nussinov. 2009. Multiresolution community detection for megascale networks by information-based replica correlations. Phys. Rev. E 80 (1) (2009) 016109

[20] A. Arenas, A. Fern?ndez, S. Fortunato, S. 2008. G?mez. Motif-based communities in complex networks. J. Phys. A 41 (22) (2008) 224001.

[21] A Lazar, D Abel, T Vicsek. 2009. Modularity Measure of Networks With Overlapping Modules. IOP Publishing, Pages: 18001

[22] M Rosvall, CT Bergstrom. 2008. Maps of random walks on complex networks reveal community structure. Proc Natl Acad Sci USA 105: 1118–1123. `http://www.tp.umu.se/~rosvall/code.html`

[23] P Pons, M Latapy. 2005. Computing communities in large networks using random walks. Sci. 3733 (2005) 284–293.

[24] A. Lancichinetti, S. Fortunato, J. Kert?sz. 2009. Detecting the overlapping and hierarchical community structure in complex networks. New J. Phys. 11, 033015, 2009

[25] Santo Fortunato. 2009 Community detection in graphs. Physics Reports , 486, 75 – 174

[26] L Donetti, MA Mu?oz. 2004. Detecting network communities: a new systematic and efficient algorithm. J. Stat. Mech. P10012 (2004).