МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование)

01.03.02 Прикладная математика и информатика
(код и наименование направления подготовки)

Системное программирование и компьютерные технологии
(направленность (профиль)/специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему «Исследование социальных сетей методами визуализации графов»

Студент И.И. Мансуров
(И.О. Фамилия) (личная подпись)

Руководитель к.т.н, Т.Г. Султанов
(ученая степень, звание, И.О. Фамилия)

Консультант М.А. Четаева
(ученая степень, звание, И.О. Фамилия)

Тольятти 2020

Аннотация

Тема: Исследование социальных сетей методами визуализации графов.

Одними из основных методов анализа социальных сетей являются методы визуализация графов.

Исследование социальных сетей методами визуализация графов является актуальным и представляют научно-практический интерес.

Объектом исследования бакалаврской работы является социальная сеть.

Предметом исследования бакалаврской работы являются методы визуализации графов.

Цель выпускной квалификационной работы – исследование социальных сетей методами визуализации графов.

Методы исследования: методы теории графов, методы анализа социальных сетей.

Описаны методы анализа социальных сетей, проанализированы известные алгоритмы визуализации графов.

Проведен экспериментальный анализ социальной сети в программе Gephi.

Результаты бакалаврской работы могут быть рекомендованы для практического решения задач анализа социальных сетей методами визуализации графов.

Структура бакалаврской работы: страниц 41, рисунков 11, источников 23.

Abstract

The topic of the given graduation work is Study of social networks using graph visualization methods.

One of the main tools of analyzing social networks are graph visualization methods. The study of social networks using graph visualization methods is relevant and of scientific and practical interest.

The object of the graduation work is social networks.

The subject of study of the graduation work are graph visualization methods.

The aim of the graduation work is study of social networks using graph visualization methods.

Research methods: graph theory methods, social network analysis methods.

The methods of analysis of social networks are described, well-known graph visualization algorithms are analyzed. Experimental social network analysis in the Gephi program is carried out.

The results of the graduation work can be recommended for practical solution of problems of social network analysis.

The graduation work consists of an explanatory note on 41 pages including 11 figures, the list of 23 references.

Оглавление

Введение

Социальная сеть – это социальная структура, состоящая из набора социальных акторов (физических лиц или организаций), наборов диадических связей и других социальных взаимодействий между участниками [21].

Яркими примерами социальных сетей являются Facebook, Twitter, ВКонтакте и др.

Рост информации в социальных сетях обусловил необходимость в создании более совершенных методах их исследования. Современная социальная сеть – это междисциплинарная область, которая объединяет математиков, ИТ-специалистов, экономистов и социологов.

SNA (Social network analysis, анализ социальных сетей) – направление современной компьютерной социологии, которое занимается описанием и анализом возникающих в ходе социального взаимодействия и коммуникации связей (сетей) различной плотности и интенсивности.

Исследование социальных сетей основано на методах анализа, использующих локальные и глобальные паттерны данных структур для определения местоположения ключевых акторов и изучения динамики сети.

Одними из таких методов являются методы визуализация графов.

Исследование социальных сетей методами визуализация графов является **актуальным** и представляет научно-практический интерес.

Объектом исследования бакалаврской работы является социальная сеть.

Предметом исследования бакалаврской работы являются методы визуализации графов.

Цель выпускной квалификационной работы – исследование социальных сетей методами визуализации графов.

Для достижения данной цели необходимо выполнить следующие задачи:

– проанализировать методы анализа социальных сетей на основе теории графов;

– проанализировать модели и алгоритмы визуализации социальной сети;

– выполнить экспериментальный анализ социальной сети с помощью методов визуализации графов.

Методы исследования – методы теории графов, методы анализа социальных сетей.

Практическая значимость бакалаврской работы заключается в исследовании возможностей существующего программного обеспечения для анализа социальных сетей методами визуализации графов.

Данная работа состоит из введения, трех глав, заключения и списка используемой литературы.

Первая глава посвящена исследованию методов анализа социальной сети на основе теории графов.

Во второй главе проанализированы модели и алгоритмы визуализации графов социальной сети.

Третья глава посвящена исследованию возможностей существующего программного обеспечения для анализа социальных сетей методами визуализации графов.

В заключении описываются результаты выполнения выпускной квалификационной работы.

Глава 1 Методы анализа социальной сети на основе теории графов

1.1 Методология анализа социальных сетей на основе теории графов

Современная наука анализа социальных сетей характеризуется четырьмя определяющими свойствами [16]:

1. В ней воплощены идеи о важности социальных связей, связывающих социальных субъектов.

2. Она собирает данные, отражающие эти связи.

3. Это предполагает использование графических изображений.

4. Используются математические и / или вычислительные модели.

Перечислим наиболее популярные задачи анализа социальных сетей [8]:

– анализ поведения пользователей: выявление аккаунтов-дубликатов, пользователей нарушающих, склонных нарушать правила, не похожих на других;

– прогнозирование: поведения пользователей (когда будет пользоваться услугами, в какую группу вступит, с кем подружится), предсказание и предотвращение ухода пользователей, предсказание трафика (в каком объёме будет скачивать/закачивать);

– рекомендация: предсказание эффективности действия рекламы для конкретного пользователя, формирование таргетированных предложений (рекламы, по вступлению в группы, заполнению профиля и т.п.)

– кластеризация: разбиение пользователей на группы (для более корректного A/B-тестирования, разработки стратегий под группы, более тщательного анализа аудитории), выявление «кругов общения пользователей» (друзей, которых объединяет некоторая сущность, например «друзья по вузу»), выделение сообществ, выделение базисов источников информации в блогосфере;

– взаимодействие с другими соцсетями/ресурсами: матчинг сетей/графов (установление соответствия между пользователями одной сети и другой), использование данных соцсети для решения задач других заказчиков;

– визуализация: поиск закономерностей в данных соцсети и их представление анализ общественного мнения по постам; научные исследования графов соцсетей и др.

Для моделирования социальных сетей используются методы теории графов.

Граф представляет собой важную сложную сетевую модель, описывающую отношения между различными объектами в реальных приложениях, включая граф знаний, социальную сеть и сеть трафика.

Соответственно, анализ социальных сетей (SNA) рассматривает социальные отношения с точки зрения теории сетей, состоящей из узлов и связей (также называемых ребрами или связями) [19].

Узлы – это отдельные субъекты внутри сетей, а связи - это отношения между акторами. Результирующие структуры на основе графа часто очень сложны. Между узлами может быть много видов связей.

Исследования в ряде академических областей показали, что социальные сети работают на многих уровнях, от семей до уровня наций, и играют решающую роль в определении способа решения проблем, организации работы и степени успеха отдельных людей. в достижении своих целей.

В своей простейшей форме социальная сеть – это карта определенных связей, таких как дружба, между изучаемыми узлами.

На рисунке 1 представлен пример графа простой социальной сети: узлы представляют людей или акторов, а грани между узлами представляют некоторые отношения между акторами.

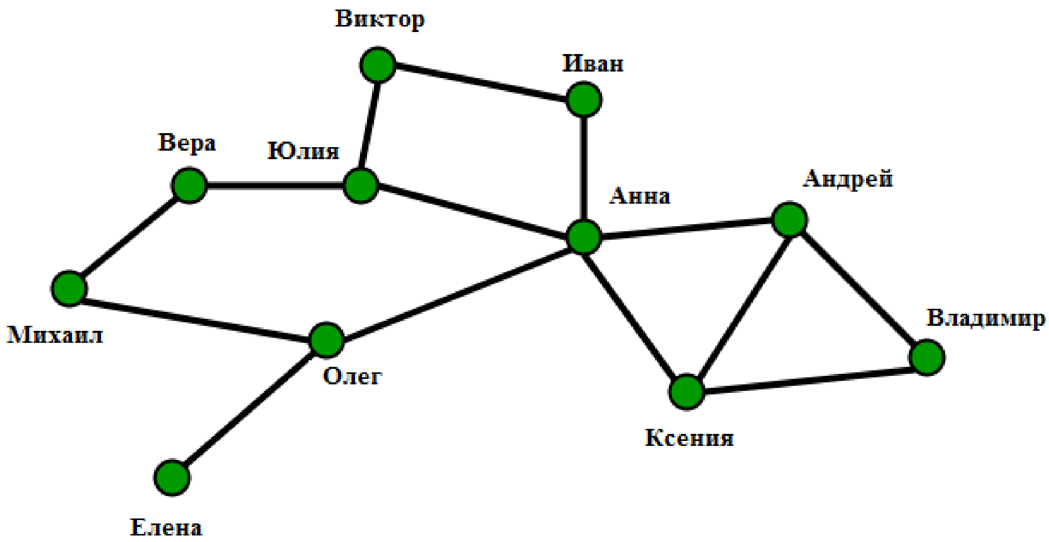


Рисунок 1 – Пример графа социальной сети

Таким образом, узлы, с которыми таким образом связан индивид, являются социальными контактами этого индивида. Сеть также может использоваться для измерения социального капитала - ценности, которую человек получает от социальной сети.

Эти понятия часто отображаются на диаграмме социальной сети, где узлы - это точки, а связи - это линии.

Как было отмечено выше, социальная сеть может быть представлена в виде графа:

$$G = (V, E), \tag{1}$$

где V - набор узлов (или вершин), связанных с акторами, а $E \in V \times V$ - набор ребер, которые соответствуют их отношениям [12].

Для представления социальных сетей используются два типа графов:

– однорежимный граф.

Это имеет место, например, в классическом наборе данных, связанном с со спортивным клубом, где узлы соответствуют членам клуба, ребра используются для описания их взаимоотношений.

Когда отношения имеют направление, ребра заменяются дугами (ориентированный граф). Узлы также, как и ребра могут иметь атрибуты. В этом случае граф считается помеченным;

– двухрежимный граф.

Если рассматриваются отношения между двумя типами элементов, например, членами и соревнованиями в спортивном клубе, для представления такой социальной сети используется двухрежимный граф.

Двухрежимный граф, также известный как двудольный граф, является графом с двумя типами вершин. Ребра разрешены только между узлами разных типов.

Наиболее распространенный способ хранения двухрежимных данных:

– прямоугольная матрица данных с двумя типами узлов соответственно в строках и столбцах. Например, двумерная матрица с акторами в строках и событиями в столбцах может представлять двухрежимный граф для спортивного клуба. Это представление очень распространено в социальных сетях. Следует отметить, что двухрежимные графы могут быть преобразованы в однорежимные с помощью проекции на один тип узла и создания ребер между этими узлами с использованием различных функций агрегирования.

Понятие графа можно обобщить с помощью гиперграфа, в котором два набора вершин могут быть соединены ребром, и мультиграфа, которому разрешено иметь ребра с одинаковыми конечными узлами.

Рассмотрим основные методы анализа социальных сетей, основанные на визуализации графов.

1.2 Анализ центральности

В рамках теории графов и сетевого анализа существуют различные меры центральности вершины в графе, которые определяют относительную важность вершины в графе (например, насколько важен человек в социальной сети или в теории пространственного синтаксиса, насколько важна комната в здании или насколько хорошо используется дорога в городской сети) [1].

1.2.1 Центральность по степени

Анализ центральности по степени **(**Degree centrality**)** является самым простым.

Центральность по степени определяется как число связей, приходящихся на узел (т.е. количество связей, которые имеет узел).

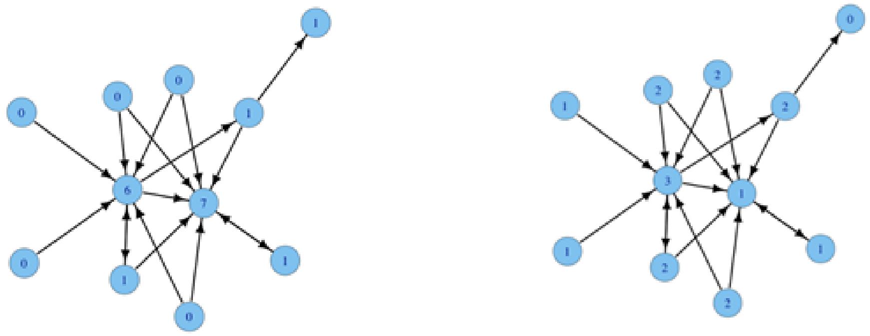
Степень часто интерпретируется с точки зрения непосредственного риска того, что узел может перехватить то, что проходит через сеть (например, вирус или некоторая информация).

Если сеть направлена (имеется в виду, что связи имеют направление), то обычно определяют две отдельные меры центральности степени:

- внутренние связи - это число связей, направленных на узел;
- внешние связи - количество связей, которые узел направляет другим.

Для позитивных отношений, таких как дружба или совет, обычно интерпретируют внутренние связи как форму популярности, а внешние связи – как общительность.

На рисунке 2 представлены примеры графов, узлы которых нагружены внутренними (а) и внешними связями (б) [13].



а)   б)

Рисунок 2 – Примеры графов, с узлами, нагруженные показателями степени центральности

Для графа, имеющего n вершин центральность по степени для вершины v определяется по формуле:

$$C_D(v) = \frac{deg(v)}{n-1}, \tag{2}$$

Временная сложность вычисления центральности по степени для всех узлов V в графе плотной матрицей смежности составляет $\Theta(V^2)$, а для всех ребер E в графе с разреженной матрицей смежности – $\Theta(E)$.

1.2.2 Центральность по близости

Центральность по близости (Closeness centrality) является показателем, насколько быстро распространяется информация в сети от одного участника к остальным.

В топологии и смежных областях математики близость является одним из основных понятий в топологическом пространстве.

Интуитивно мы говорим, что два множества близко, если они произвольно близко друг к другу. Понятие может быть определено естественным образом в метрическом пространстве, где определено понятие расстояния между элементами пространства, но оно может быть обобщено на топологические пространства, где у нас нет конкретного способа измерения расстояний.

В теории графов близость является мерой центральности вершины в графе. Вершины, которые «неглубокие» по отношению к другим вершинам (то есть те, которые имеют тенденцию иметь короткие геодезические расстояния до других вершин в графе), имеют более высокую близость.

Близость предпочтительнее в сетевом анализе для обозначения длины кратчайшего пути, поскольку она дает более высокие значения большему количеству центральных вершин и поэтому обычно положительно связана с другими показателями, такими как степень (рисунок 3).

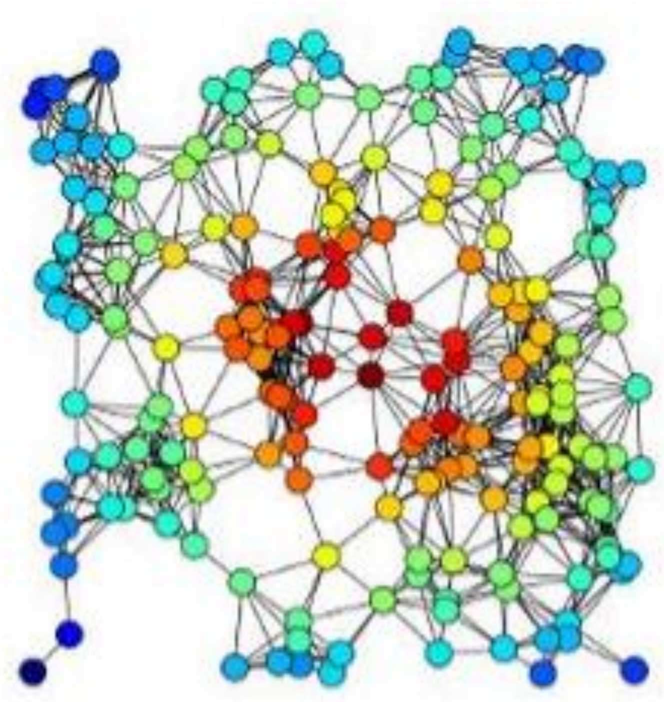


Рисунок 3 – Центральность по близости

В теории сетей близость является сложной мерой центральности. Она определяется как среднее геодезическое расстояние (то есть кратчайший путь) между вершиной v и всеми другими достижимыми вершинами:

$$C_C(v) = \frac{n-1}{\sum_{t \in V \setminus v} d_G(v,t)}, \quad (3)$$

где $d_G(v,t)$ – кратчайший путь от вершины v до вершины t.

Рассмотрение расстояния из или во все другие узлы неприменимо для неориентированных графов, тогда как в ориентированных графах они дают совершенно различные результаты.

Например, интернет-сайт может иметь высокую степень близости от исходящего соединения, но низкую степень близости от входящих соединений).

1.2.3 Посредническая центральность

Посредническая центральность (Betweenness centrality) является мерой центральности вершины в графе.

Вершины, которые встречаются на многих кратчайших путях между другими вершинами, имеют более высокую промежуточность чем те, которые этого не делают.

Для графа с n вершинами промежуточность для вершины вычисляется следующим образом:

Шаг 1. Для каждой пары вершин (s, t) вычислим все кратчайшие пути между ними.

Шаг 2. Для каждой пары вершин (s, t) определим долю кратчайшие пути, которые проходят через рассматриваемую вершину (здесь вершина v).

Шаг 3. Суммируем эту долю по всем парам вершин (s, t).

Таким образом:

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}, \tag{4}$$

где σ_{st} – общее количество кратчайших путей из вершины $\boldsymbol{s}$ к вершине $\boldsymbol{t}$;

$\sigma_{st}(v)$ – количество кратчайших путей из вершины s к вершине t, проходящих через вершину v.

Для нормализации нужно разделить на количество пар вершин, за исключением самой вершины v, т. е. для ориентированного графа нужно разделить на (n-1)(n-2), для неориентированного – на величину, равную (n-1)(n-2)/2.

На рисунке 4 представлен пример посреднической центральности [11].

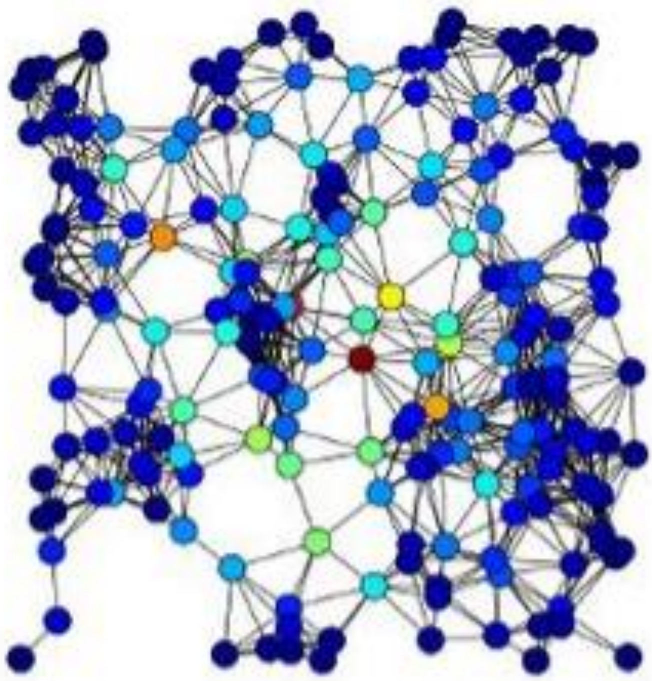


Рисунок 4 – Посредническая центральность

Недостатком посреднической центральности является ее вычислительная сложность.

1.2.4 Центральность по собственному вектору

Центральность по собственному вектору (Eigenvector centrality) является мерой важности узла в сети. Она назначает относительные оценки всем узлам в сети на основе принципа, согласно которому подключения к узлам с высокой оценкой вносят больший вклад в оценку рассматриваемого узла, чем равные подключения к узлам с низкой оценкой.

Google PageRank является вариантом меры центральности по собственному вектору [20].

Пусть x_i – оценка центральности i-го узла. Обозначим матрицу смежности сети через $A_{i,j}$.

Тогда $A_{i,j}$ =1, если i-й узел находится рядом с j-м узлом и, $A_{i,j}$ =0 – в противном случае.

В более общем случае, записи в A могут быть действительными числами, представляющими силу соединения, как в стохастической матрице.

Пусть для i-го узла оценка центральности будет пропорциональна сумме оценок всех узлов, которые к нему подключены.

Следовательно, получим:

$$x_i(t) = \frac{1}{\lambda} \sum_{j \notin M(i)} x_i = \frac{1}{\lambda} \sum_{j=1}^{N} A_{i,j} x_i, \qquad (5)$$

где M(i) – множество узлов, подключенных к i–му узлу;

N – общее количество узлов;

λ – константа.

В векторной нотации получим:

$$\mathbf{x} = \frac{1}{\lambda} \mathbf{A}\mathbf{x} \qquad (6)$$

Как правило, будет много разных собственных значений центральности λ, для которых существует решение для собственного вектора (рисунок 5).

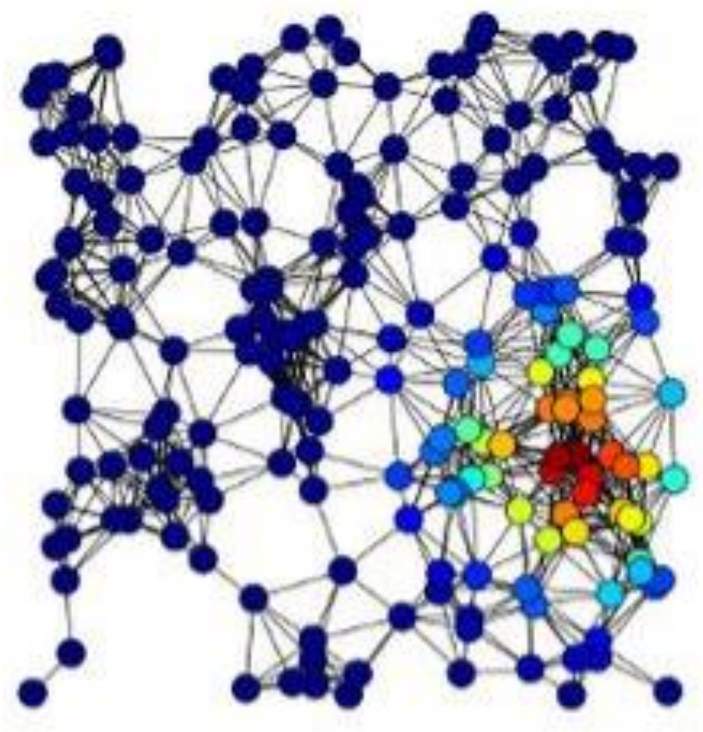


Рисунок 5 – Центральность по собственному вектору

Однако дополнительное требование, чтобы все записи в собственном векторе были положительными, подразумевает (по теореме Перрона-Фробениуса), что только наибольшее собственное значение приводит к желаемой мере центральности

Собственный вектор, соответствующий самому большому собственному значению, как раз образован центральностями соответствующих участников сети.

Таким образом, чем больше у участника друзей и чем они центральнее, тем больше его центральность. Верно и обратное: чем больше центральность участника, тем больше центральность его друзей.

Недостатком центральности по собственному вектору также является вычислительная сложность.

1.2.5 Центральность Каца

В теории графов центральность Каца является обобщенной мерой центральности в сети (рисунок 6).

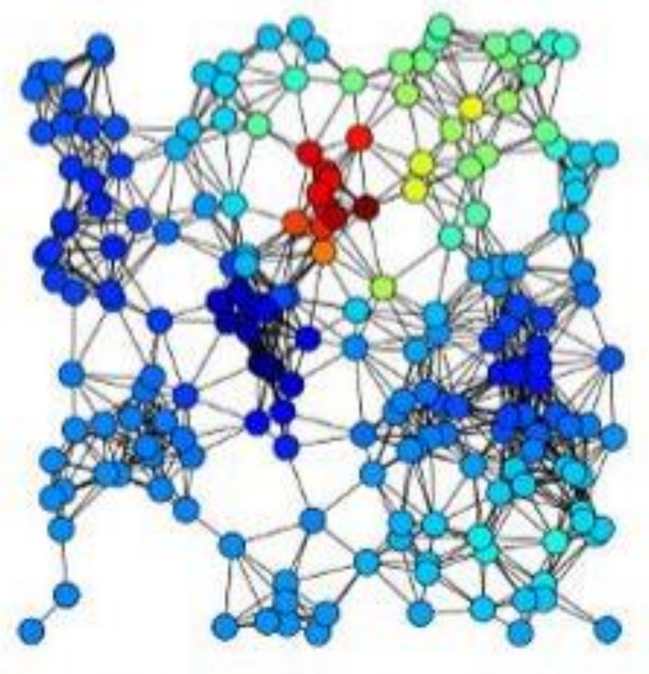


Рисунок 6 – Центральность Каца

Она был введена Лео Кацем в 1953 году и используется для измерения относительной степени влияния актера (или узла) в социальной сети [18].

В отличие от типичных мер центральности, которые рассматривают только кратчайший путь (геодезический) между парой действующих лиц, центральность Каца измеряет влияние, принимая во внимание общее количество прогулок между парой действующих лиц.

Пусть A – матрица смежности рассматриваемой сети.

Элементы a_{ij} матрицы A являются переменными, которые принимают значение 1, если узел i связан с узлом j, и 0 в противном случае.

Степени A указывают на наличие (или отсутствие) связей между двумя узлами через посредников.

Например, в матрице A^3, если элемент $a_{2,12} = 1$, это указывает, что узел 2 и узел 12 соединены через несколько соседей первой и второй степени узла 2.

В формализованном виде центральность Каца определяется следующим образом:

$$C_{\text{Katz}}(i) = \sum_{k=1}^{\infty} \sum_{j=1}^{n} \alpha^k (A^k)_{ji}, \tag{7}$$

Следует отметить, что в приведенном выше определении используется тот факт, что элемент в расположении (i,j) матрицы A смежности, возведенный в степень k (A^k), отражает общее количество степеней k соединений между узлами i и j.

Выводы к первой главе

1. Для моделирования социальных сетей используются методы теории графов. Граф представляет собой важную сложную сетевую модель, описывающую отношения между различными объектами в реальных приложениях, включая граф знаний, социальную сеть и сеть трафика.

2. Для анализа социальных сетей используются различные показатели центральности. В рамках теории графов и сетевого анализа существуют различные меры центральности вершины в графе, которые определяют относительную важность вершины в графе. Центральность по степени определяется как число связей, приходящихся на узел. Центральность по близости является показателем, насколько быстро распространяется информация в сети от одного участника к остальным. Посредническая центральность является мерой центральности вершины в графе. Центральность по собственному вектору является мерой важности узла в сети. Она назначает относительные оценки всем узлам в сети на основе принципа, согласно которому подключения к узлам с высокой оценкой вносят больший вклад в оценку рассматриваемого узла, чем равные подключения к узлам с низкой оценкой. Обобщенной мерой центральности узла в сети в теории графов является центральность Каца.

3. Анализ центральности по степени является самым простым. Недостатком других показателей центральности является их вычислительная сложность.

Глава 2 Модели и алгоритмы визуализации социальной сети в виде графов

2.1 Постановка задачи визуализации социальной сети в виде графа

Визуализация является одной из самых востребованных функций в программах обработки графов, и это относится и к программному обеспечению сетевого анализа.

Введем понятие укладки графа.

Рассмотрим обыкновенные (неориентированные, без петель и кратных ребер) графы [9].

Под укладкой графа G (V, E) понимается отображение вершин графа в множество точек d-мерного пространства:

$$L : V \to R^d \tag{8}$$

Как показывает практика, визуализация является одним из преимуществ использования графов для анализа социальных сетей.

Визуализация графа представляет собой процедуру указания координат его вершин и изображения ребер графа в виде отрезков между вершинами.

Позицию вершины v_i обозначим через p_{vi} или p_i, ее координаты – через $p_i^1,\ldots, p_i^d$.

Под расстоянием между вершинами v и u понимается евклидово расстояние между соответствующими точками, которое обозначим через $\|p_v - p_u\|$.

В задаче динамической визуализации требуется нарисовать последовательность графов G_1, G_2, …, G_T, описывающих некоторые данные за последовательные промежутки времени.

С каждой вершиной $v \in G$ ассоциирована метка l_v.

$l_v = l_u$ тогда и только тогда, когда вершины v и u соответствуют одному объекту исходных данных.

2.2 Модели визуализации графов

Рассмотрим задачу:

Дано: граф G (V, E).

Необходимо создать четкий и читаемый рисунок графа G.

Критерии оптимизации:

- соседние узлы должны быть размещены рядом;
- несмежные узлы должны быть размещены на удалении друг от друга;
- ребра короткие, прямые, одинаковой длины;
- плотно связанные части (кластеры) образуют сообщества;

- как можно меньше переходов;
- узлы распределены равномерно.

Для решения данной задачи используются силовые (*force-directed*) модели укладки графов.

Силовые модели укладки графов используются для преобразования математической информации графа в двумерную или трехмерную геометрическую информацию, основанную на относительном положении узлов и ребер, автоматического вытягивания графа и, таким образом, реализации его визуализации [23].

В настоящее время существует множество алгоритмов, основанных на силовой модели укладки графа: прямая линия, метод прямоугольника, метод сетки, круговой метод, метод дисконтирования основаны на математических методах, в соответствии с математической характеристикой графики они получают графические координаты вершин по уравнению.

Рассмотрим принципы силовой модели укладки графов.

2.2.1 Пружинная модель

Идея пружинной модели построена на предположении, что каждая вершина графа представляет собой шарик. Так как оптимальные координаты шарика не известны, шарик случайно размещен в двумерных плоскостях. Между шарами имеется пружина. Идеальная длина пружин пропорциональна расстоянию между шариками. Если расстояние между шариками не равно идеальной длине пружин, пружина в системе будет растянута или сжата, и система будет накапливать энергию.

Если энергия системы не равна нулю, шары будут двигаться из-за сил притяжения и отталкивания. Как только шары перестают двигаться, энергия системы становится равной нулю, и система переходит в состоянии равновесия. В этот момент координатное положение шарика находится в оптимальном положении в структуре вершин графа.

Опишем математическую модель пруженной системы.

Введем следующие определения:

1. x (i) представляет координаты узла i;

2. d (i, j) представляет идеальную длину пружины между узлами i и j (теоретическое расстояние между узлами i и j);

3. ||x (i) - x (j) || – фактическое расстояние между узлами i и j;

Тогда энергия пружины между узлами i и j может быть описана следующим образом:

$$\text{En} = (\|x(i) - x(j)\| - d(x, j))^2 \qquad (9)$$

Как следует из уравнения, энергии пружины, если расстояние между узлами i и j равно идеальной длине пружин, энергия пружины будет равна 0.

Если расстояние между узлами i и j больше идеальной длины пружины (пружина растянута), или расстояние между узлами i и j меньше идеальной длины пружины (пружина сжата), энергия пружины будет больше 0. Чем больше различий между фактическим расстоянием между узлами и предполагаемой длиной пружины, тем больше будет энергия пружины.

Таким образом, энергия пружинной системы представляет собой сумму энергий пружин между всеми узлами, а именно:

$$\text{Energy} = \sum\nolimits_{i \neq j} (\|x(i) - x(j)\| - d(x, j))^2 \qquad (10)$$

Таким образом, построение оптимальной структуры эквивалентно математической задаче оптимизации без ограничений.

Это означает, что надо найти координату x (i) всех узлов, используя минимальную энергию. Координата x(i) в этом случае является оптимальным решением задачи.

Сила каждой вершины может быть рассчитана следующим образом.

Если расстояние между узлами i и j больше идеальной длины пружины (пружина растянута), вершина i будет притягиваться пружиной с силой:

$$\mathrm{AF(v)} = (\|x(i) - x(j)\| - d(x, j) * ((x(j) - x(i))) / (\|x(i) - x(j)\| \qquad (11)$$

Если расстояние между узлами i и j меньше идеального длина пружины (пружина сжата), вершина i будет отталкиваться пружиной с силой:

$$\mathrm{RF(v)} = (\|x(i) - x(j)\| - d(x, j) * ((x(j) - x(i))) / (\|x(i) - x(j)\|$$

(12)

На самом деле две силы одинаковы, только имеют различные направления.

Таким образом, сила F (i) вершины i может быть описана следующим образом:

$$F(i) = \sum_j (\|x(i) - x(j)\| - d(x, j)) * ((x(j) - x(i))) / \|x(i) - x(j)\| \qquad (13)$$

Алгоритм, основанный на пружинной модели состоит из следующих шагов:

Шаг 1 примем начальный шаг t = 1

Шаг 2: вершина случайным образом размещена в двумерном (или трехмерном) пространстве, координата вершины i равна x (i)

Шаг 3: рассчитать полную силу F (i) каждой вершины i

Шаг 4: вычислить изменения положения (d x (i) = t * F (i)) каждой вершины i

Шаг 5: если все изменения положения d x (i) малы, вывести x и остановиться

Шаг 6: обновить x и t: x (i) = x (i) + d x (i), t = 0,9 * t

Шаг7: перейти на шаг3.

Недостаток силового алгоритма, основанного на пружинной модели, заключается в том, что он должен вычислять теоретико-графическое расстояние между любыми двумя вершинами.

2.2.2 Пружинная электрическая модель

Чтобы решить проблему, состоящую в том, что пружинная модель отнимает много времени при расчете теоретико-графического расстояния, создан силовой алгоритм, основанный на пружинной электрической модели.

Идея электрической модели пружины такова: предположим, что вершины – это шары, которые имеют одинаковый заряд и отталкиваются друг от друга, причем сила отталкивания обратно пропорциональна геометрическому расстоянию между шарами.

Если две вершины соединены ребром, соответствующие шары будут связаны пружиной, идеальная длина которой равна 0.

Если сила отталкивания шаров равна силе притяжения, энергия всей системы будет равна нулю, и система перейдет в состояние равновесия. В противном случае энергия всей системы не будет равна 0, и система не будет в равновесном состоянии.

Шар не будет двигаться под воздействием сил притяжения и отталкивания до тех пор, пока суммарные силы равны нулю.

Пусть:

1. x (i) представляет координаты вершины i;

2. ‖ x (i) -x (j) ‖ представляет фактическое расстояние между вершинами i и j.

Тогда электронные силы отталкивания между любыми вершинами i и j составляют:

$$\mathrm{RForce} = -CK^2 / \|x(i) - x(j)\| \tag{14}$$

То есть силы отталкивания обратно пропорциональны расстоянию: чем ближе расстояние, тем больше будет сила отталкивания.

Соответственно, сила притяжения между вершинами i и j, которые не связаны ребром:

$$\mathrm{AForce} = \|x(i) - x(j)\|^2 / K \tag{15}$$

То есть силы притяжения пропорциональны квадрату расстояния: чем больше расстояние, тем больше притяжение.

Если сила отталкивания шара равна силе притяжения, энергия всей системы будет равна нулю, иначе, чем больше сила каждой вершины, тем выше энергия всей системы.

Таким образом, задача оптимальной структуры тогда эквивалентна проблеме минимальной энергии. Аналогично, эта проблема оптимизации также может применять итеративный алгоритм, который является тем же в предыдущей модели, чтобы найти оптимизацию.

В пружинной модели вводится только пружина, приводящая к физическому явлению, которое легко понять и облегчает математические вычисления, поскольку вычисление теоретического расстояния является более сложным, чем вычисление геометрического расстояния, поэтому в тех же итерационных условиях алгоритмы основаны на алгоритмах.

Вместе с тем, алгоритмы на основе пружинной электрической модели более эффективны, чем алгоритмы, основанные на пружинной модели.

2.3 Алгоритмы визуализации графов

1. Пружинный алгоритм Идса

Алгоритм был разработан Идсом (Eads) в 1984 г. [14].

Псевдокод алгоритма имеет вид:

Вход: $G = (V, E)$ - связный неориентированный граф с начальным размещением $p = (p_v)_{v \in V}$, количество итераций $K \in N$, порог $\varepsilon > 0$, константа > 0

Выход: Укладка графа p с «низким внутренним стрессом.

$t \leftarrow 1$

while $t < K$ **and** $\max_{v \in V} \|F_v(t)\| > \varepsilon$ **do**

for each v∈V **do**

$$F_v(t) \leftarrow \sum_{u:\{u,v\}\in E} f_{rep}(p_u, p_v) + \sum_{u:\{u,v\}\in E} f_{spring}(p_u, p_v)$$

for each v∈V **do**

$p_v \leftarrow p_u + \delta * F_v(t)$

t ← t+1

2. Алгоритм Камада-Кавайи

Псевдокод алгоритма имеет вид:

Установить начальные скорости узла (0,0).

Произвольно установить начальные положения узлов // убедиться, что 2 узла не находятся в одной и той же позиции

цикл

total kinetic energy: = 0 // полная кинетическая энергия, текущая сумма общей кинетической энергии по всем частицам для каждого узла.

net-force: = (0. 0) // текущая сумма общей силы на этом конкретном узле.

для каждого следующего узла

net-force: = net-force + Кулоновское_отталкивание (this_node, other_node)

для каждой пружины, связанной с узлом

net-force := net-force + Сила_Гука(this node, spring) next

пружина

this node.velocity: = (this_node.velocity + time-step net-force) * damping

this node.position: = this_node.position + timestep * this_node.velocity

total kinetic energy: = total_kinetic_energy + this_node.mass (this_node.velocity) ^ 2

следующий узел

пока total kinetic energy меньше некоторого значения

Пример результата визуализации по алгоритму Камада-Кавайи приведен на рисунке 7.

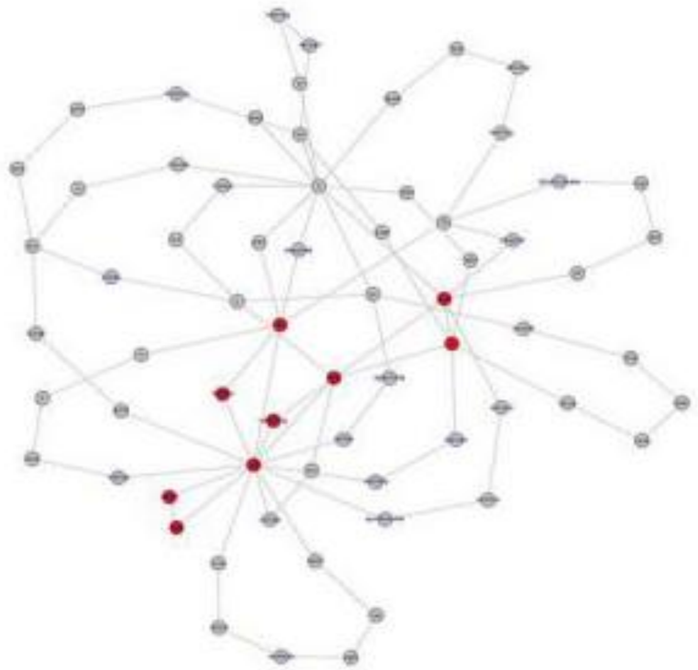


Рисунок 7 - Результат визуализации графа по алгоритму Камада-Кавайи

3. Алгоритм **Фрутермана-Рейнгольда**

Псевдокод алгоритма имеет вид:

```
for i := 1 to max_iterations do begin
for each v in Vertices do begin {вычисление силы
        отталкивания}
        v.pos' := (0,0)
        for each u in Vertices do
                if (u != v) then begin
                        Delta := v.pos - u.pos;
                    v.pos' := v.pos' + (Delta / len(Delta) ) * f_r(len(Delta))
                end
        end
 for each e in Edges do begin {вычисление силы притяжения}
        Delta := e.start.pos' - e.end.pos';
```

```
        e.start.pos' := e.start.pos' + (Delta / len(Delta) ) * f_a(len(Delta));
        e.end.pos' := e.end.pos' + (Delta / len(Delta) ) * f_a(len(Delta));
    end
    for v in Vertices do begin {предельное смещение}
        Delta := v.pos' - v.pos;
        v.pos := v.pos + (Delta / len(Delta) ) * min(len(Delta), t);
end
```

Пример результата визуализации по алгоритму Фрутермана-Рейнгольда приведен на рисунке 8.

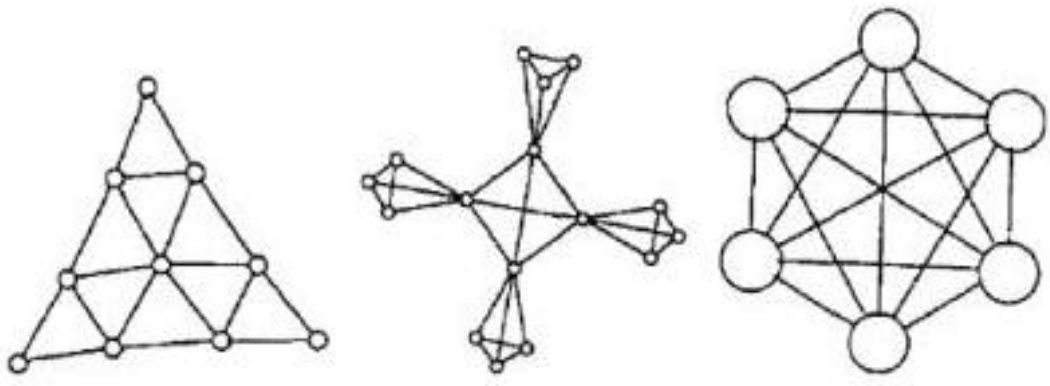


Рисунок 8 - Результаты визуализации графов по алгоритму Фрутермана-Рейнгольда

Как показал анализ, общим недостатком указанных алгоритмов является недостаточно разработанные критерии их сходимости.

Выводы ко второй главе

1. Исследование социальных сетей основано на методах анализа, использующих локальные и глобальные паттерны данных структур для определения местоположения ключевых акторов и изучения динамики сети.

2. Силовые модели укладки графов используются для преобразования математической информации графа в двумерную или трехмерную

геометрическую информацию, основанную на относительном положении узлов и ребер, автоматического вытягивания графа и реализации его визуализации.

3. Алгоритмы на основе пружинной электрической модели более эффективны, чем алгоритмы, основанные на пружинной модели.

4. Общим недостатком указанных алгоритмов является недостаточно разработанные критерии их сходимости.

Глава 3 Программное обеспечение для анализа социальных сетей методами визуализации графов

Программы для визуализации графов можно разделить на две категории:

1) редакторы — приложения, которые позволяют автоматизировать ручное построение графа и обеспечивают возможность автоматического изменения граней при перемещении вершин графа;

2) программы, которые позволяют отказаться от ручного рисования графа. При этом пользователь задает некоторую базу данных, по которой программа рассчитывает положение вершин и граней и строит граф. Для построения социограмм могут применяться программы обоих типов, поскольку для конкретных случаев та или иная программа может оказаться наиболее подходящей [compress].

Одной из программ, относящихся ко второй категории, используемых для анализа социальных сетей, является программа Gephi .

Gephi — ведущее программное обеспечение для визуализации и исследования всех видов графиков и сетей. Gephi распространяется с открытым исходным кодом и бесплатно [22].

Программа работает на ОС Windows, Mac OS X и Linux (рисунок 9).

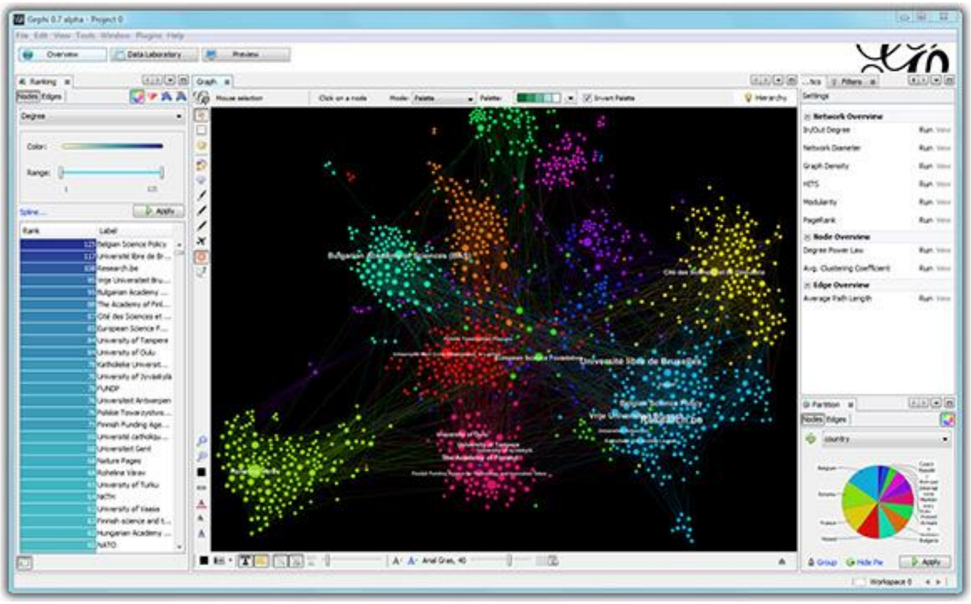


Рисунок 9 – Главное окно программы Gephi

Анализ социальных сетей обеспечивается простым созданием коннекторов социальных данных для сопоставления общественных организаций и небольших сетей.

Поддерживает вычисление всех показателей центральности.

Используемые технологии:

- эргономичный интерфейс: не требуются навыки программирования;
- высокая производительность: встроенный движок рендеринга;
- собственные форматы файлов: GDF (GUESS), GraphML (NodeXL), GML, NET (Pajek), GEXF и другие;
- настраивается с помощью плагинов: макеты, метрики, источники данных, инструменты манипуляции, пресеты рендеринга и многое другое.

Выполним экспериментальный анализ групп социальных сетей с помощью программы Gephi.

В качестве примера рассмотрим анализ групп новостных изданий.

Ниже представлен код визуализации графа на языке Python [2].

```
%matplotlib inline
import networkx
import requests
import json
def getVKMembers(group_id, count=1000, offset=0):
# http://vk.com/dev/groups.getMembers
host = 'http://api.vk.com/method'
if count > 1000:
raise Exception('Bad params: max of count = 1000')
response
=requests.get('{host}/groups.getMembers?group_id={group_id}&count={co
unt}&offset={offset}'
.format(host=host, group_id=group_id, count=count, offset=offset))
if not response.ok:
raise Exception('Bad response code')
return response.json()
def allCountOffset(func, func_id):
```

```
set_members_id = set()
count_members = -1
offset = 0
while count_members != len(set_members_id): # posible endless loop for real vk api
response = func(func_id, offset=offset)['response']
if count_members != response['count']:
count_members = response['count']
new_members_id = response['users']
offset += len(new_members_id)
if set_members_id | set(new_members_id) == set_members_id != set(): # without new members
print 'WARNING: break loop', count_members, len(set_members_id)
break
set_members_id = set_members_id.union(new_members_id)
return set_members_id
groups = ['http://vk.com/meduzaproject',
'http://vk.com/tj',
'http://vk.com/smmrussia',
'http://vk.com/vedomosti',
'http://vk.com/kommersant_ru',
'http://vk.com/kfm',
'http://vk.com/oldlentach',
'http://vk.com/lentaru',
'http://vk.com/lentasport',
'http://vk.com/fastslon',
'http://vk.com/tvrain',
'http://vk.com/sport.tvrain',
'http://vk.com/silverrain',
'http://vk.com/afishagorod',
```

```
'http://vk.com/afishavozduh',
'http://vk.com/afishavolna',
'http://vk.com/1tv',
'http://vk.com/russiatv',
'http://vk.com/vesti',
'http://vk.com/ntv',
'http://vk.com/lifenews_ru']
members = {}
for g in groups:
name = g.split('http://vk.com/')[1]
print name
members[name] = allCountOffset(getVKMembers, name)
matrix = {}
for i in members:
for j in members:
if i != j:
matrix[i+j] = len(members[i] & members[j]) * 1.0/ min(len(members[i]),
len(members[j]))
max_matrix = max(matrix.values())
min_matrix = min(matrix.values())
for i in matrix:
matrix[i] = (matrix[i] - min_matrix) / (max_matrix - min_matrix)
g = networkx.Graph(directed=False) for i in members:
for j in members:
if i != j:
g.add_edge(i, j, weight=matrix[i+j])
members_count = {x:len(members[x]) for x in members}
max_value = max(members_count.values()) * 1.0
size = [] max_size = 900
min_size = 100
```

```
for node in g.nodes():
size.append(((members_count[node]/max_value)*max_size + min_size)*10)
import matplotlib.pyplot as plt
pos=networkx.spring_layout(g)
plt.figure(figsize=(20,20))
networkx.draw_networkx(g, pos, node_size=size, width=0.5, font_size=8)
plt.axis('off')
plt.show()
```

В результате выполнения в программе Gephi получим следующий граф (рисунок 10).

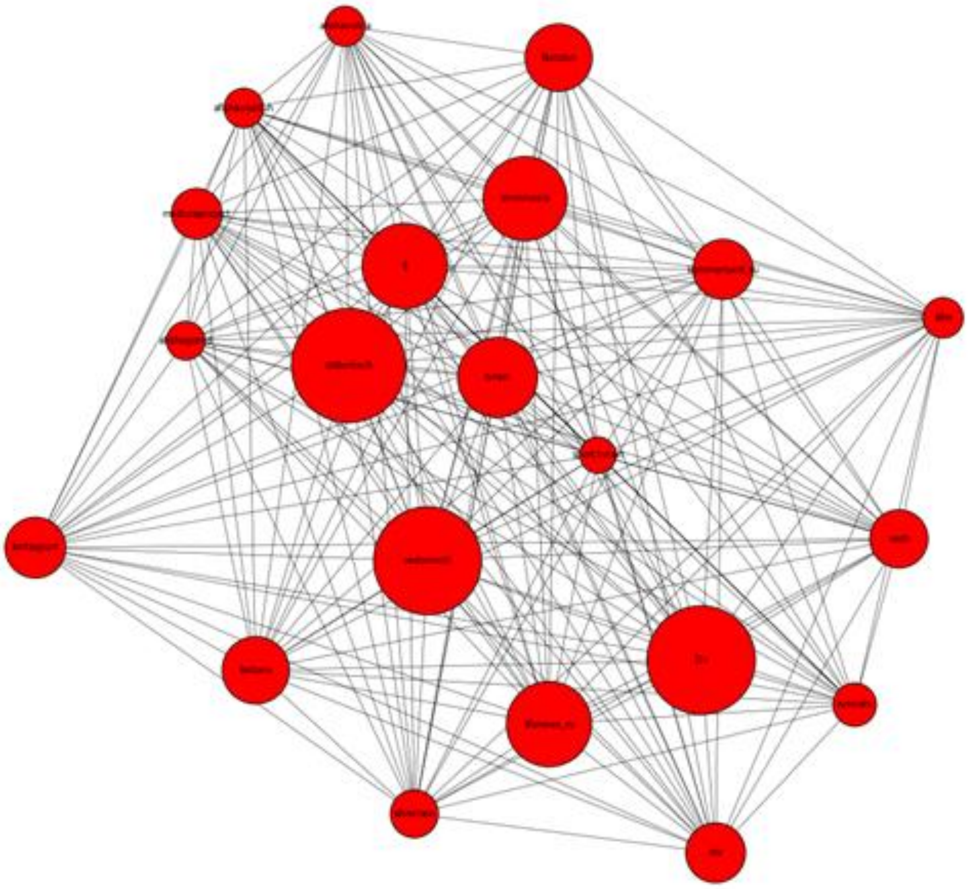


Рисунок 10 – Граф новостной группы

Промежуточность – число присутствия вершины в кратчайших путях между любыми другими вершинами.

Исследование блогосферы с помощью программы Gephi показало, что высокой степенью промежуточности обладает крайне небольшое количество узлов – всего 6 или около 0.5% [3].

Это означает, что в политическом сегменте Рунета не наблюдается сложной разветвленной сети со множеством больших кластеров и сообществ.

Как правило, пользователи-проводники информации имеют возможность передавать информацию, общаясь одновременно в 2-4 различных кругах политических мнений. При этом эти проводники информации не обладают большой влиятельностью на мнение сообществ, в которых состоят, поэтому затруднительно использовать их в информационных кампаниях в предвыборный период.

На рисунке представлен граф 11, в котором наибольшим размером и цветом теплых оттенков (зеленый, оранжевый и красный) выделены пользователи, обладающие наибольшей степенью промежуточности.

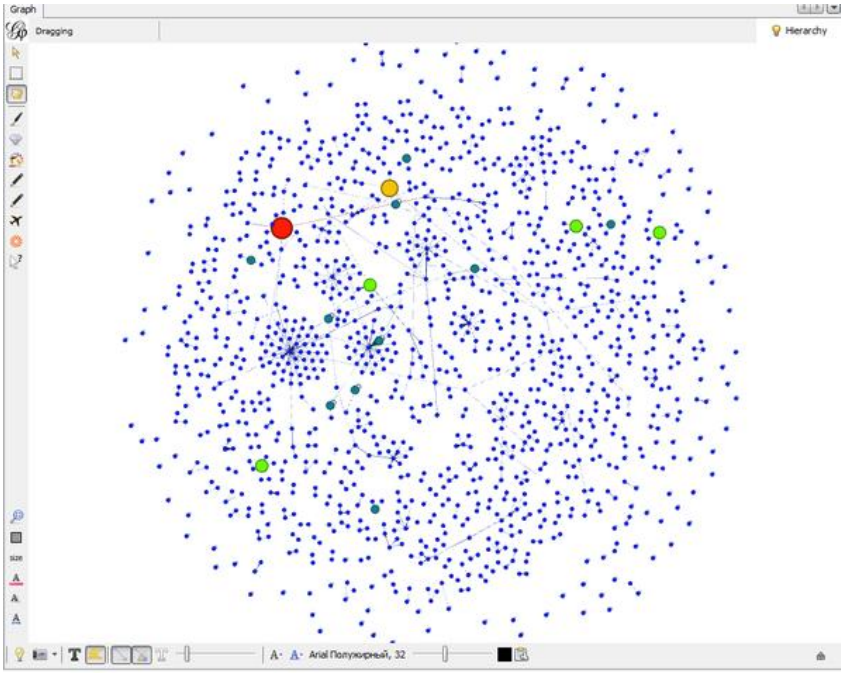


Рисунок 11 – Граф с выделенными вершинами с высокой степенью промежуточности

Как показало экспериментальное тестирование, программа Gephi позволяет успешно решать задачи анализа социальных сетей с помощью методов визуализации графов.

Выводы к третьей главе

1. Программы для визуализации графов можно разделить на две категории: редакторы — приложения, которые позволяют автоматизировать ручное построение графа программы, которые позволяют отказаться от ручного рисования графа. Ко второй категории программ относится программа Gephi.

2. Анализ социальных сетей в Gephi обеспечивается простым созданием коннекторов социальных данных для сопоставления общественных организаций и небольших сетей.

3. Как показало экспериментальное тестирование, программа Gephi позволяет успешно решать задачи анализа социальных сетей с помощью методов визуализации графов.

Заключение

Одними из основных методов анализа социальных сетей являются методы визуализация графов.

Исследование социальных сетей методами визуализация графов является актуальным и представляют научно-практический интерес.

Целью бакалаврская работы является исследование социальных сетей методами визуализации графов.

Для достижения поставленной цели были решены следующие задачи:

1. Проанализированы методы анализа социальных сетей на основе теории графов. Для анализа социальных сетей используются различные показатели центральности. В рамках теории графов и сетевого анализа существуют различные меры центральности вершины в графе, которые определяют относительную важность вершины в графе: центральность по степени, центральность по близости, посредническая центральность, центральность по собственному вектору является мерой важности узла в сети. Обобщенной мерой центральности узла в сети в теории графов является центральность Каца. Анализ центральности по степени является самым простым. Недостатком других показателей центральности является их вычислительная сложность.

2. Описаны и проанализированы модели и алгоритмы визуализации социальной сети в виде графа. Как показал анализ, силовые модели укладки графов используются для преобразования математической информации графа в двумерную или трехмерную геометрическую информацию, основанную на относительном положении узлов и ребер, автоматического вытягивания графа и реализации его визуализации. Алгоритмы на основе пружинной электрической модели более эффективны, чем алгоритмы, основанные на пружинной модели. Общим недостатком указанных алгоритмов является недостаточно разработанные критерии их сходимости.

3. Как показал анализ, программы для визуализации графов можно условно разделить на две категории: редакторы — приложения, которые позволяют автоматизировать ручное построение графа программы, которые позволяют отказаться от ручного рисования графа. Ко второй категории программ относится программа Gephi.

4. С помощью программы Gephi выполнен экспериментальный анализ социальной сети методами визуализации графов. Анализ социальных сетей в Gephi обеспечивается простым созданием коннекторов социальных данных для сопоставления общественных организаций и небольших сетей. Как показало экспериментальное тестирование, программа Gephi позволяет успешно решать задачи анализа социальных сетей с помощью методов визуализации графов.

Результаты бакалаврской работы представляют научно-практический интерес и могут быть рекомендованы для практического решения задач анализа социальных сетей методами визуализации графов.

Список используемой литературы и используемых источников

1. Батура Т. В. Модели и методы анализа компьютерных социальных сетей // Программные продукты и системы. 2013. №3 (103). С. 130-137.

2. Введение в анализ социальных сетей на примере VK API [Электронный ресурс]. URL: https://habr.com/ru/post/263313/ (дата обращения: 13.06.2020).

3. Визуализация графа социальной сети: анализ событий блогосферы перед декабрём 2011 года [Электронный ресурс]. URL: https://habr.com/ru/post/164307/ (дата обращения: 13.06.2020).

4. ГОСТ 19.402–78. Единая система программной документации. Описание программы.

5. ГОСТ 19.701-90 (ИСО 5807-85) Единая система программной документации (ЕСПД). Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения.

6. Долинина О. Н., Печенкин В. В., Тарасова В. В. Подходы к динамической визуализации графов социальных сетей образовательной организации // Вестник СГТУ. 2011. №4 (62). С. 239-242

7. Компьютерная визуализация социальных сетей [Электронный ресурс]. URL: https://compress.ru/article.aspx?id=16593#%D0%9A%D1%80%D0%B0%D1%82%D0%BA%D0%B8%D0%B9%20%D0%BE%D0%B1%D0%B7%D0%BE%D1%80%20%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%20%D0%B4%D0%BB%D1%8F%20%D0%B2%D0%B8%D0%B7%D1%83%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8%20%D1%81%D0%BE%D1%86%D0%B8%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D1%85%20%D1%81%D0%B5%D1%82%D0%B5%D0%B9 (дата обращения: 20.05.2020).

8. Прикладные задачи анализа данных: анализ социальных сетей [Электронный ресурс]. URL: http://www.machinelearning.ru/wiki/images/e/e7/PZAD2016_14_social.pdf (дата обращения: 20.05.2020).

9. Пупырев С. Н., Тихонов А. В. Визуализация динамических графов для анализа сложных сетей // Моделирование и анализ информационных систем. 2010. Т.17. № 1. С. 117–135.

10. A Force-Directed Algorithm for Drawing Directed Graphs Symmetrically [Электронный ресурс]. URL: https://www.hindawi.com/journals/mpe/2018/6208509/alg2/ (дата обращения: 20.05.2020).

11. Betweenness centrality [Электронный ресурс]. URL: https://www.ebi.ac.uk/training/online/course/network-analysis-protein-interaction-data-introduction/building-and-analysing-ppins-3 (дата обращения: 20.05.2020).

12. Combe D., Largeron C. Egyed-Zsigmond E., Géry M. A comparative study of social network analysis tools. Web intelligence and virtual enterprises (Saint Etienne. France), 2010.

13. Degree Centrality [Электронный ресурс]. URL: https://www.sci.unich.it/~francesc/teaching/network/degree.html (дата обращения: 20.05.2020).

14. Eades P. A Heuristic for Graph Drawing // Congressus Numerantium. 1984. Vol 42 (11).

15. Force-Directed Graph Layout [Электронный ресурс]. URL: https://www.yworks.com/pages/force-directed-graph-layout#:~:text=A%20force%2Ddirected%20graph%20drawing,organic%20and%20aesthetically%20pleasing%20way (дата обращения: 20.05.2020).

16. Freeman L.C. Social Network Visualization, Methods of. In: Meyers R. (eds) Computational Complexity. Springer, NY, 2012.

17. Kamada T., Kawai S. An algorithm for drawing general undirected graphs // Information Processing Letters. 1989. Vol. 31(1).

18. Katz Centrality (Centrality Measure) [Электронный ресурс]. URL: https://www.geeksforgeeks.org/katz-centrality-centrality-measure/ (дата обращения: 20.05.2020).

19. Kosorukoff A. Social network analysis: Theory and applications, 2011.

20. Page Rank checker [Электронный ресурс]. URL: https://www.prchecker.info/check_page_rank.php (дата обращения: 20.05.2020).

21. Social network [Электронный ресурс]. URL: https://en.wikipedia.org/wiki/Social_network (дата обращения: 20.05.2020).

22. The Open Graph Viz Platform [Электронный ресурс]. URL: https://gephi.org/ (дата обращения: 20.05.2020).

23. Wang Lingling, Wang Xianshui, Wang Qiuquan and u Mei. Research on Force-directed Algorithm Optimization Methods // Proc. of International Conference on e-Education, e-Business and Information Management (ICEEIM 2014), 2014. P. 13-17.