

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**ПРИМЕНЕНИЕ МЕТОДОВ РАНЖИРОВАНИЯ УЗЛОВ ГРАФА К
ПРИКЛАДНЫМ ЗАДАЧАМ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студента 4 курса 451 группы
направления 09.03.04 — Программная инженерия
факультета КНиИТ
Колесникова Виктора Анатольевича

Научный руководитель
доцент, к. ф.-м. н.

С. В. Миронов

Заведующий кафедрой
к. ф.-м. н.

С. В. Миронов

Саратов 2019

ВВЕДЕНИЕ

Компания Google создала PageRank для решения проблемы своей поисковой системы, позволяющей выдавать набор веб-страниц, содержащих практически те же слова, что и введенный пользователем запрос. Разработчики Google хотели включить меру важности страницы в эти результаты, чтобы отличить хорошо известные и релевантные страницы от тех, которые были менее известны. Для этого Google сформировал систему ранжирования страниц под названием PageRank, которая использовала структуру ссылок в интернете, чтобы определить, какие страницы являются более важными.

Целью данной дипломной работы является формирование представления о применимости методов и алгоритмов анализа графов, построенных на основе данных из специфичных прикладных задач, для нахождения наиболее важных объектов в этих областях. Поставлены требования изучения способов формирования графов по специфике прикладной задачи. В зависимости от предметной области между всеми объектами, участвующими в ранжировании, расставляются отношения, соединяющие эти объекты, благодаря чему образуется граф, способный быть оцененным различными метриками. Так же необходимо изучить теоретический материал алгоритмов центральности графов и методов ранжирования и возможности их реализации на практике. Для формирования истинного представления об эффективности данного подхода необходимо научиться сравнивать полученные результаты с истинными на тестовых данных.

Цель бакалаврской работы – разработать программный продукт, позволяющий оценить степень эффективности методов оценки графов к трем следующим задачам:

- ранжирование коллекции документов на основе неполной индексации,
- составление рейтинга противоборствующих сторон, соревнующихся в наборе турниров,
- анализ пищевой цепи Национального парка «Хвалынский».

Структура и объём работы. Бакалаврская работа состоит из введения, 3 разделов, заключения, списка использованных источников и 9 приложений. Общий объем работы – 90 страниц, из них 40 страниц – основное содержание, включая 11 рисунков и 5 таблиц, список использованных источников информации – 23 наименований.

1 Прикладные задачи

В современное время развитие методов хранения и передачи информации привело к значительному увеличению ее общего объема. Справляться с ранжированием этих коллекций становится все труднее используя ручное сопоставление каждого из документов с каждым, по-этому разрабатываются и внедряются автоматизированные алгоритмы. В данной работе ставится задача применения мер ранжирования узлов графа для поиска наиболее значимых вершин в наборе из трех специфичных прикладных задач.

1.1 Задача ранжирования объектов на основе неполной индексации

Первой является задача оценки публикаций на научно-исследовательских или учебно-образовательных конкурсах, где количество присланных работ может быть намного больше времени отведенного для выступлений. При этом необходимо выбрать лучшие из них для последующего оценивания. Здесь необходимо разбить все документы по пачкам и распределить их между некоторым количеством рецензентов, и на основе их оценки составить и обработать граф.

1.2 Формирование рейтинга снукеристов

В качестве одной из прикладных задач было принято взять проблему формирования топ-рейтинга профессиональных игроков в игру снукер за 2018/2019 календарный год [1]. Необходимо проанализировать все проведенные крупные турниры в этом промежутке и на основе пар победитель — проигравший составить результирующий граф. Его отличительной особенностью будет возможность наличия нескольких дуг с разной направленностью между вершинами, так как игроки могут встречаться друг с другом несколько раз и не обязательно один игрок всегда будет одерживать победу над другим.

1.3 Биологические сети

Одной из задач настоящей работы является анализ экосистемы смешанного леса Национального парка «Хвалынский» [2]. По данным, полученным о пищевой цепи данной локации, необходимо определить наиболее важные виды животных. Пищевую цепь представим в виде невзвешенного ориентированного графа, где вершинами будут служить виды, а дугами пары кто кем питается.

2 Технологии и алгоритмы

2.1 Меры центральности

Одной из первых в теории сетевого анализа разрабатывалась идея центральности. Мер, позволяющих оценить центральность, множество. Каждая из них связана с определенным алгоритмом. В контексте данной работы будут рассмотрены нижеописанные метрики.

2.1.1 Closeness centrality

Центральность по близости (closeness centrality) выражает, насколько близко узел расположен к остальным узлам сети. Формально центральность по близости выражается как отношение числа других узлов графа к сумме кратчайших расстояний между определенным узлом и всеми другими [3]. Если какая-то вершина недостижима из данной расстояние до нее считается равной общему количеству вершин в графе. Так, центральность по близости C_C для вершины x неориентированного графа $G = (V, E)$, где V — множество вершин, $|V| = n$, E — множество ребер, вычисляется по формуле:

$$C_C(x) = \frac{1}{\sum_{y \in V} d(y, x)}, \quad (1)$$

где $d(y, x)$ кратчайшее расстояние от вершины y до x . Если они не связны, то $d(y, x) = n$.

2.1.2 Betweenness centrality

Центральность по посредничеству (betweenness centrality) — характеризует, насколько важную роль данный узел играет на пути между другими узлами. Центральность по посредничеству выражает, сколько кратчайших путей между всеми узлами сети проходит через определенный узел.

Пусть задан граф $G = (V, E)$, где V — множество вершин, E — множество ребер. Обозначим число вершин как $n = |V|$, а число ребер как $m = |E|$. Тогда центральность вершины $v \in V$ определяется следующей формулой:

$$C_B(v) = \frac{1}{n_B} \sum_{\substack{s, t \in V \\ s \neq v \neq t}} \frac{\sigma_{s,t}(v)}{\sigma_{s,t}}, \quad (2)$$

где $\sigma_{s,t}$ — число геодезических (кратчайших) путей между вершинами $s \in V$ и

$t \in V$, $\sigma_{s,t}(v)$ — число геодезических путей между вершинами $s \in V$ и $t \in V$, проходящих через вершину v .

2.2 PageRank

PageRank — один из алгоритмов ссылочного ранжирования. Алгоритм применяется к коллекции документов, связанных гиперссылками, и назначает каждому из них некоторое численное значение, измеряющее его «важность» или «авторитетность» среди остальных документов.

Спецификой данного параметра является то, что он высчитывается в зависимости от веса ссылочной массы. Простыми словами: PR — это суммарный авторитет страницы, который напрямую зависит от всех ссылаемых на него сайтов и конечного получения общего веса этих ссылок. Именно вес всех ссылок и будет определять такое значение, как PageRank. Следует отметить, что PR учитывает не количество ссылок, а именно их вес — так что огромное количество заспамленных ссылок не сделают картину.

Теория PageRank утверждает, что воображаемый пользователь, который случайно переходит по ссылкам в конце концов перестанет их нажимать. Вероятность на любом этапе, что пользователь будет продолжать является коэффициентом затухания d . В различных исследованиях были протестированы различные коэффициенты затухания, но обычно предполагается, что его значение будет установлен около $d = 0.85$.

В совокупности получаем уравнение, выглядящее следующим образом:

$$PR(p_i) = \frac{1 - d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}, \quad (3)$$

где p_1, p_2, \dots, p_n рассматриваемые страницы, $M(p_i)$ набор страниц, которые ссылаются на p_i , $L(p_j)$ количество исходящих из p_j ссылок, N общее количество страниц, d коэффициент затухания.

Для нахождения PR можно воспользоваться итерационным методом решения уравнений, так как запись рангов всех страниц может быть представлена в виде квадратной системы линейных алгебраических уравнений [4].

3 Эмпирическая часть

3.1 Ранжирование документов

Для анализа применимости методов, позволяющих оценивать графы разных структур и вершины в них, к задаче ранжирования коллекции документов в порядке улучшения их качества, сформулируем и реализуем нижеописанную модель.

3.1.1 Распределение по пачкам

В условиях наличия большого количества объектов N с неизвестным отношением порядка, подлежащих ранжированию, и недостаточного количества рецензентов K , такого что $N \gg K$, не способных сделать соответствие каждого из документа коллекции с каждым для получения полной модели отношения порядка, ставится задача разбить все объекты по пачкам (*batch*) и распределить их между ограниченным количеством рецензентов.

Количество документов в пачке определим числом p , ограниченным в интервале $[4; 10]$. А число пачек, отправляемых на рецензирование каждому эксперту, обозначим через переменную n .

Так как пачки будут составляться произвольно, путем последовательного взятия p случайных документов из всей коллекции, по окончании распределения произойдет смещение увеличения встречаемости к определенным документам, в свою очередь как некоторые вовсе могут не попасть ни в одну пачку. Чтобы избавиться от этого изъяна, реализуем функцию `__min_incidence(self)` для нахождения множества документов с минимальной встречаемостью, чтобы уже из него случайным образом выбирать документы для пачек.

После того, как стало возможно формирование пачек с равномерным показателем вхождений для всех документов из коллекции, осталось распределить их между всеми экспертами.

3.1.2 Отношение порядка в пачке

Когда все пачки сформированы и распределены в словаре `distrib` между всеми рецензентами, встает задача оценки документов внутри них. В рамках данной работы будем рассматривать два вида оценок: безошибочная оценка и с генерацией ошибки. Необходимо выставить отношение порядка между каждыми парами документов в пачке.

3.1.3 Безошибочная оценка

При генерировании идеального варианта работы рецензентов с нулевой вероятностью ошибки при расставлении отношения порядка в графе, будем придерживаться предположение того, что реальное отношение порядка для всей коллекции документов выглядит как отсортированный лист номеров всех документов: $[0, 1, 2, \dots, N - 2, N - 1]$. Следовательно, чем меньший номер имеет документ в пачке, тем он будет и более высоко оценен рецензентами.

3.1.4 Генерация ошибки оценки

При необходимости формирования условий, когда рецензенты делают ошибки, необходимо определиться с какой вероятностью будут появляться неверные отношения порядка. Для этой цели была разработана формула выдачи правильного ответа с вероятностью $prob$ для двух документов i и j с соответствующими номерами doc_i и doc_j в истинном отношении порядка:

$$Probability(i, j) = \begin{cases} prob + (doc_i - doc_j) / (\frac{1}{1-prob} N), & doc_i > doc_j, \\ (1 - prob) + (doc_i - doc_j) / (\frac{1}{1-prob} N), & doc_i < doc_j. \end{cases} \quad (4)$$

3.1.5 Измерение сходства графов

Основываясь на предположение, что истинное отношение порядка для коллекции документов соответствует убыванию порядковых номеров, можно дать оценку полученному результату ранжирования. Для этого необходимо построить матрицы $R = [r_{ij}]$ для оцениваемых списков следующим образом:

$$r_{ij} = \begin{cases} 1, & doc_i > doc_j, \\ 0, & doc_i < doc_j. \end{cases} \quad (5)$$

где doc_i порядковый номер документа i в ранжированном списке.

Расстояние между двумя матрицами R_{ideal} и R_{actual} , принимающие только значения 0 и 1, можно оценить с помощью формулы расстояния Хэмминга:

$$d(R_{ideal}, R_{actual}) = \frac{\sum_i^n \sum_{j \neq i}^n |r_{ij}^{ideal} - r_{ij}^{actual}|}{n(n-1)}, \quad (6)$$

где n — размерность квадратной матрицы [5].

Таблица 1 – Показатели зависимости ранжирования методом closeness centrality от наполненности графа из 1000 документов при отсутствующей ошибке рецензентов

Наполненность	Средняя ошибка	Минимальная	Максимальная
0.59	13.22	11.92	14.64
1.20	8.63	8.06	9.36
2.48	4.82	4.50	5.14
4.41	2.70	2.57	2.84
8.64	1.31	1.20	1.36
11.89	0.90	0.83	0.95
24.51	0.35	0.34	0.37
43.03	0.14	0.13	0.15
59.05	0.07	0.07	0.7
71.65	0.04	0.04	0.04
91.80	0.01	0.01	0.01
99.45	0.0004	0.0004	0.0004

3.1.6 Ранжирование графа

Для того, чтобы сформировать представление о зависимости качества оценки ранжирования от наполненности графа введем величину $g_{completeness}$ для графа $G = (V, E)$, где V – множество вершин, E – множество ребер вычисляемую по формуле:

$$g_{completeness} = \frac{|E|}{|V|(|V| - 1)/2}. \quad (7)$$

3.1.7 Closeness centrality

Для получения полного представления о ранжировании с помощью центральности по близости произведем последовательно несколько последовательных экспериментов. Рассчитаем у графа показатель closeness centrality [6] для каждой вершины, отсортируем их в порядке убывания и сравним с истинным отношением порядка. Изменяя количество пачек, отправляемых рецензентам, и количество документов в них, можно изменять уровень заполненности результирующего графа, для прослеживания зависимости качества ранжирования от уровня наполненности, как при безошибочной оценке рецензентов (см. Табл. 1), так и с ошибкой в выставлении отношений порядка в пачках (см. Табл. 2).

Таблица 2 – Показатели зависимости ранжирования методом closeness centrality от наполненности графа из 1000 документов при ошибке рецензирования $prob = 5\%$

Наполненность	Средняя ошибка	Минимальная	Максимальная
0.59	25.63	24.56	26.77
1.20	17.61	17.13	18.19
1.99	13.41	12.91	13.86
4.42	9.59	9.29	10.08
8.12	6.67	6.43	6.88
10.05	6.64	6.33	7.04
18.02	7.49	7.30	7.77
37.18	3.67	3.52	3.82
56.74	2.43	2.39	2.48
85.33	1.95	1.94	1.96
96.45	1.84	1.82	1.85

3.1.8 Betweenness centrality

Суть алгоритма центральности по посредничеству [7] не подходит к сформированному графу, так как находит ключевые узлы по которым проходит наибольшее количество кратчайших путей, поэтому вне зависимости от наполненности графа ошибка будет стремиться к 50%.

3.1.9 PageRank

По аналогии для двух видов оценивания найдем результаты ранжирования pagerank в зависимости от уровня наполненности (см. Табл. 3-4).

3.2 Формирование рейтинга снукеристов

Для нахождения топа лучших игроков в снукер необходимо проанализировать данные о всех крупных турнирах за 2018/2019 календарный игровой год. Всего было проведено 23 соревнования, в которых приняли участия 124 снукериста.

3.2.1 Формирование исходных данных

Запишем каждый турнир в текстовый документ, отведенный под него, в виде пар, соответствующих турнирной сетке соревнования. Пара представлена в виде имен игроков и количества фреймов, выигранных ими в партии. Победитель пары определяет, в каком направлении будет выставлена дуга в

Таблица 3 – Показатели зависимости ранжирования методом PageRank от наполненности графа из 1000 документов при отсутствующей ошибке рецензентов

Наполненность	Средняя ошибка	Минимальная	Максимальная
0.60	13.31	12.80	13.89
1.20	9.29	9.01	9.59
1.78	7.34	6.95	7.85
2.38	6.42	6.02	6.69
2.96	4.00	3.77	4.19
4.89	4.28	4.09	4.42
7.25	3.37	3.27	3.46
10.67	2.71	2.62	2.82
18.42	1.94	1.91	1.98
28.76	1.49	1.41	1.56
39.20	1.16	1.12	1.19
57.13	0.82	0.78	0.85
78.54	0.48	0.46	0.50
95.22	0.18	0.18	0.19

Таблица 4 – Показатели зависимости ранжирования методом PageRank от наполненности графа из 1000 документов при ошибке рецензирования $prob = 5\%$

Наполненность	Средняя ошибка	Минимальная	Максимальная
0.60	20.95	19.69	22.77
1.20	18.74	17.89	19.72
2.38	18.61	18.07	19.42
4.89	17.44	16.60	18.80
7.25	17.02	16.74	17.59
10.73	16.28	15.57	16.76
18.59	14.77	13.94	15.25
29.23	12.96	12.72	13.13
40.14	11.66	11.42	11.92
59.37	10.04	9.55	10.39
78.53	8.99	8.83	9.18
92.9	8.66	8.38	8.89
99.31	8.96	8.58	9.07

графе турнира, от игрока с меньшим количеством выигранных фреймов (проигравшего) к игроку с большим количеством (победителю).

Для формирования графа, построенного на основании проведения большого количества соревнований, был реализован класс `Snooker`. Данный класс имеет методом инициализации `_init_(self, directory)`, принимающий в качестве параметра путь до директории, содержащей информацию об анализируемых турнирах.

3.2.2 Обработка графа

После того как на основе набора соревнований был построен результирующий граф, его можно оценить мерами центральности. Зная принцип формирования его структуры, исключим из обработки алгоритм `betwenness centrality`, потому что нас не интересуют промежуточные звенья.

Применив алгоритмы `closseness centrality` и `pagerank`, получим топ лист, соответствующий этим мерам ранжирования. Для доказательства эффективности применения методов анализа графов к задаче формирования общего рейтинга сторон, соревнующихся в большом наборе турниров, сравним полученные результаты с мировым рейтингом [8]. Приведя топ-20 игроков из каждого рейтинга (см. Табл. 5), мы можем наблюдать в них сильное сходство.

3.3 Нахождение наиболее значимых видов в пищевой цепи

3.3.1 Входные данные

Для нахождения наиболее важных видов животных были получены данные, основанные на пищевой цепи смешанного леса Национального парка «Хвалынский», представленные в следующей форме (см. Рис. 1), где каждому виду поставлен в соответствие свой уникальный номер, и множество того, чем данный вид питается.

3.3.2 Визуализация цепи

Для наглядности полученных результатов реализуем функцию `draw(self, g)`, позволяющую отображать продуцентов, консументов 1-го, 2-ого и 3-его порядков и редуцентов на своих фиксированных нишах (см. Рис. 2).

Таблица 5 – Топ-20 игроков мирового рейтинга и результаты алгоритмов closeness centrality и pagerank

№	World Rating	Closeness	PageRank
1	Ронни О'Салливан	Джадд Трамп	Кайрен Уилсон
2	Джадд Трамп	Кайрен Уилсон	Джадд Трамп
3	Марк Уильямс	Ронни О'Салливан	Стюарт Бинэм
4	Нил Робертсон	Нил Робертсон	Нил Робертсон
5	Джон Хиггинс	Стюарт Бинэм	Ронни О'Салливан
6	Марк Селби	Дэвид Гилберт	Джек Лисовски
7	Марк Аллен	Джек Лисовски	Джон Хиггинс
8	Кайрен Уилсон	Джон Хиггинс	Дэвид Гилберт
9	Барри Хокинс	Марк Аллен	Марк Аллен
10	Дин Джуньху	Барри Хокинс	Марк Селби
11	Джек Лисовски	Стюарт Бинэм	Аллистер Картер
12	Дэвид Гилберт	Райан Дэй	Барри Хокинс
13	Стюарт Бинэм	Стивен Магуайр	Стивен Магуайр
14	Шон Мерфи	Марк Селби	Райан Дэй
15	Лука Бресель	Аллистер Картер	Скотт Дональдсон
16	Стивен Магуайр	Дин Джуньху	Марк Дэвис
17	Аллистер Картер	Джо Перри	Джо Перри
18	Райан Дэй	Шон Мерфи	Шон Мерфи
19	Джо Перри	Марк Уильямс	Дин Джуньху
20	Гэри Уилсон	Марк Дэвис	Лу Хаотянь

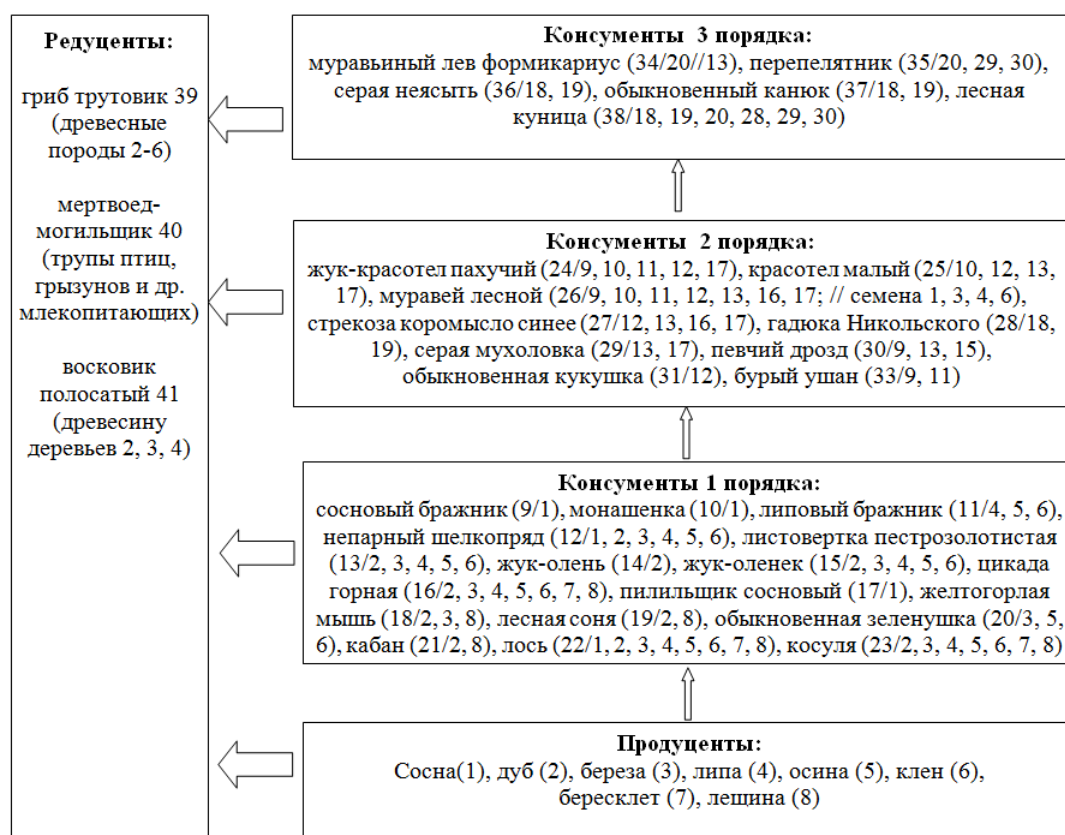


Рисунок 1 – Смешанный лес Национального парка «Хвалынский»

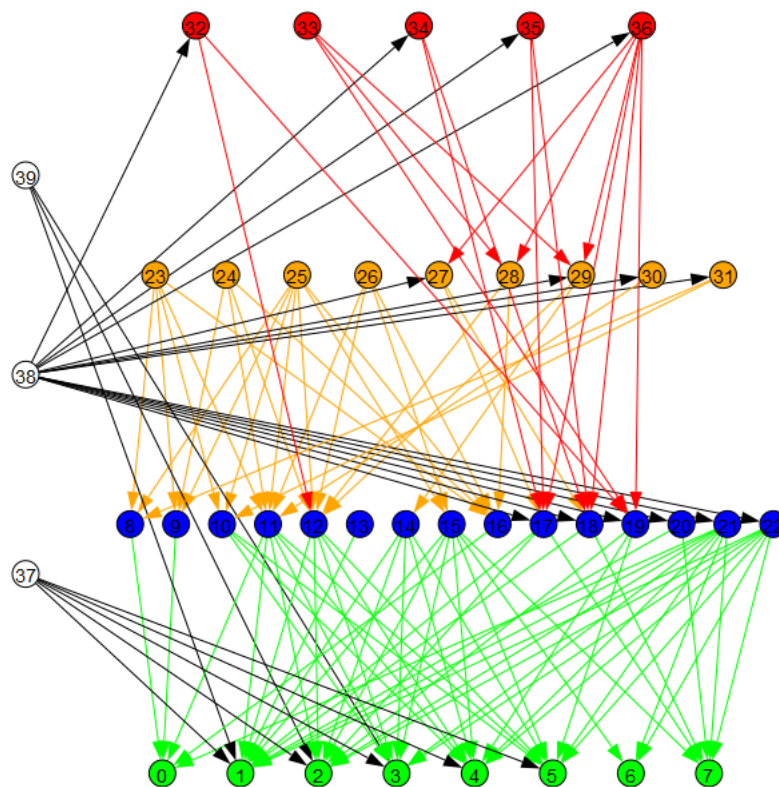


Рисунок 2 – Представление полной пищевой цепи

3.3.3 Closeness centrality

Для нахождения центральности вершины по близости предварительно реализуем обход графа в ширину, потому что он позволяет найти кратчайшие пути до остальных вершин в невзвешенном графе. Для нахождения closeness centrality от всех вершин была написана функция `check_closeness(self, g)`. В ней были сформированы и отсортированы все показатели центральности по близости как собственной функции `self.closeness centrality(i)` в список `w1` так и импортированной из `igraph` в список `w2`.

Список `w1` выглядит следующим образом: [1, 2, 4, 5, 3, 0, 7, 12, 16, 11, 6, 17, 18, 19, 8, 10, 14, 9, 29, 28, 15, 27, 20, 21, 22, 30, 31, 32, 34, 35, 36, 13, 23, 24, 25, 26, 33, 37, 38, 39].

Так как нас в большей степени интересует значимость животных в сети, а не растений, найдем первые три вершины являющиеся консументами: 12 — листовертка пестрозолотистая, 16 — пильщик сосновый, 11 — непарный шелкопряд. Удалим их из системы, что привело за собой к исчезновению еще двух видов животных, являющимися консументами второго порядка: 28 — серая мухоловка и 30 — обыкновенная кукушка, и так же к исчезновению 31 связи.

3.3.4 Betweenness centrality

Для нахождения *betweenness centrality* от всех вершин была написана функция `check_betweenness(self, g)`. В ней были сформированы все показатели центральности по близости как собственной функции в список `w1` так и импортированной из `igraph` в список `w2`.

Список `w1` выглядит следующим образом: [12, 11, 17, 19, 15, 28, 29, 10, 18, 16, 14, 21, 22, 31, 36, 8, 30, 9, 32, 20, 0, 1, 2, 3, 4, 5, 6, 7, 13, 23, 24, 25, 26, 27, 33, 34, 35, 37, 38, 39]. По аналогии с центральностью по близости найдем первые три вершины являющиеся консументами: 12 — листовертка пестрозолотистая, 11 — непарный шелкопряд, 17 — желтогорлая мышь. Удалим их из системы, что повлечет за собой к исчезновению еще одного вида животного, являющимся консументом второго порядка, 30 — обыкновенная кукушка, и так же к исчезновению 31 связи.

3.3.5 PageRank

Нахождение PageRank каждой вершины графа реализуем в функции `page_rank(self)` с помощью итерационного метода решения системы линейных уравнений. Коэффициент затухания установим в теоретически рекомендуемую величину $d = 0.85$.

Для визуализации и проверки PageRang всех вершин была написана функция `check_pagerank(self, g)`. По аналогии были составлены два списка: как собственной функции `self.pagerank()` `w1`, так и импортированной из `igraph` функции `g.pagerank()` в `w2`.

Список `w1` выглядит следующим образом: [0, 1, 2, 4, 5, 7, 3, 12, 17, 18, 11, 16, 19, 8, 10, 14, 9, 29, 28, 6, 15, 27, 20, 21, 22, 30, 31, 32, 34, 35, 36, 13, 23, 24, 25, 26, 33, 37, 38, 39]. Найдем первые три вершины являющиеся консументами: 12 — листовертка пестрозолотистая, 17 — желтогорлая мышь, 18 — лесная соня. Удалим их из системы, что повлекло за собой к исчезновению еще трех видов животных, являющимися консументами второго и третьего порядка: 27 — гадюка Никольского, 34 — серая неясыть, 35 — обыкновенный канюк, и так же к исчезновению 30 связей.

ЗАКЛЮЧЕНИЕ

В результате выполнения дипломной работы были выполнены все поставленные задачи.

На тестовых данных была доказана применимость алгоритмов оценок центральности вершин в графе к задаче ранжирования коллекции документов при неполной индексации. Алгоритмы closeness centrality и pagerank показали эффективную оценку при малом заполненности результирующего графа и оказались довольно устойчивыми при моделировании условий допущения ошибок экспертами. Также при увеличении заполненности графа при отсутствующей ошибке рецензентов, отклонение результатов алгоритмов от истинного отношения порядка быстро стремится к нулю.

При формировании рейтинга снукеристов за 2018/2019 соревновательный календарный год мерами центральности, было показано девяносто процентное совпадение первой двадцатки мирового рейтинга с результатом алгоритма closeness centrality и восьмидесяти пяти процентное совпадение с результатом алгоритма ссылочного ранжирования pagerank. Данные показатели подтверждают жизнеспособность данного подхода при формировании рейтинга соперничающих сторон на основе набора турниров.

Были найдены наиболее важные виды в пищевой цепи по трем мерах центральности вершины в графе:

- closeness centrality: 12 — листовертка пестрозолотистая, 16 — пильщик сосновый, 11 — непарный шелкопряд;
- betweenness centrality: 12 — листовертка пестрозолотистая, 11 — непарный шелкопряд, 17 — желтогорлая мышь;
- pagerank: 12 — листовертка пестрозолотистая, 17 — желтогорлая мышь, 18 — лесная соня.

Частичное пересечение результатов трех разных мер лишней раз доказывает верность проведенных вычислений по нахождению наиболее значимых видов в смешанном лесу Национального парка «Хвалынский».

Отдельные части бакалаврской работы были опубликованы на конференции: VII Международная молодёжная научно-практическая конференция «Математическое и компьютерное моделирование в экономике, страховании и управлении рисками» [9].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 World Snooker [Электронный ресурс]. — URL: <http://www.worldsnooker.com/> (Дата обращения 26.05.2019). Загл. с экр. Яз. англ.
- 2 Национальный парк «Хвалынский» [Электронный ресурс]. — URL: <http://nphvalynskiy.ru/> (Дата обращения 26.05.2019). Загл. с экр. Яз. рус.
- 3 Анализ социальных сетей/Ключевые понятия [Электронный ресурс]. — URL: http://letopisi.org/index.php/Анализ_социальных_сетей/Ключевые_понятия (Дата обращения 26.05.2019). Загл. с экр. Яз. рус.
- 4 What Is Google PageRank [Электронный ресурс]. — URL: <https://searchengineland.com/what-is-google-pagerank-a-guide-for-searchers-webmasters-11068> (Дата обращения 26.05.2019). Загл. с экр. Яз. англ.
- 5 *Faizliev, A. Analysis of the dynamics of company co-mention network and market network based on graph similarity measures / A. Faizliev, V. Balash // Future Internet. — 2019. — P. 5.*
- 6 Closeness centrality of vertices [Электронный ресурс]. — URL: <https://igraph.org/r/doc/closeness.html> (Дата обращения 26.05.2019). Загл. с экр. Яз. англ.
- 7 Vertex and edge betweenness centrality [Электронный ресурс]. — URL: <https://igraph.org/r/doc/betweenness.html> (Дата обращения 26.05.2019). Загл. с экр. Яз. англ.
- 8 Рейтинг игроков. Сезон 2018/2019 :: Снукер [Электронный ресурс]. — URL: http://world-snooker.ru/rankings_2018_2019.html (Дата обращения 26.05.2019). Загл. с экр. Яз. рус.
- 9 VII Международная молодёжная научно-практическая конференция [Электронный ресурс]. — URL: <http://risk2018.sgu.ru/> (Дата обращения 26.05.2019). Загл. с экр. Яз. рус.