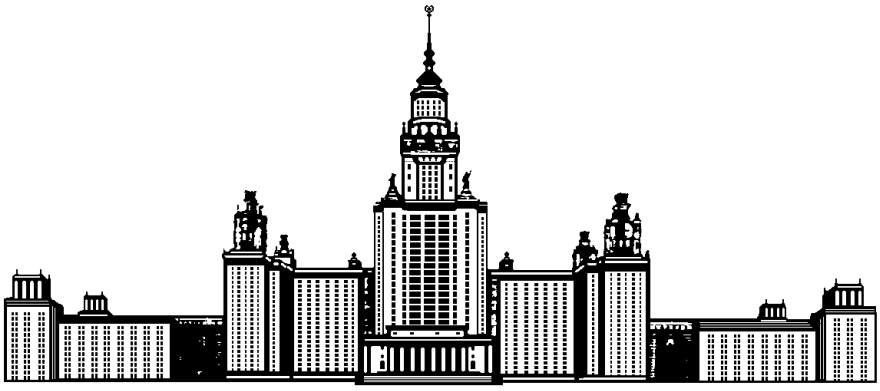


Московский государственный университет имени М. В. Ломоносова Факультет
вычислительной математики и кибернетики
Кафедра математических методов прогнозирования

Бобров Евгений Александрович

Специализированные методы визуализации корпусных данных

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Научный руководитель
к.ф.-м.н., доцент
Майсурадзе Арчил Иериевич

Москва, 2017

Содержание

Аннотация

В работе исследована коллекция научных статей. Проведена работа на всех трёх уровнях аналитической деятельности. Моделирование данных и их интеграция в информационную систему. Разработка аналитических пространств в целях оперативного анализа и визуализации. Сетевое моделирование и сравнение методов анализа графов сотрудничества. Также решён набор аналитических задач. Построены графы соавторства стран и ключевых слов в статьях. В допололнение к графовой кластеризации стран раскрашена карта мира.

Глава 1

Введение

Люди, работающие в самых разных областях: самолётостроении, химической промышленности, программировании, управлении и автоматизации бизнеса осознали, что занимаются одним и тем же видом деятельности – созданием точного формального описания некоторой предметной области. Необходимо было решать насущные проблемы, связанные с преодолением сложности конструирования и моделирования самолётов, сложности организации данных, сложности управления многоуровневой и активно растущей организацией, сложности классификации и структуризации огромного архива взаимосвязанных фактов.

Формальные описания всегда строились с целью автоматизировать или упростить тот или иной аспект деятельности, навести порядок, структурировать, но сохранить определённый уровень гибкости и расширяемости. Естественный язык является замечательной формальной моделью, которая объединяет множество предметных областей и позволяет записать самые разные знания, накопленные человечеством.

Однако требуется определить такую модель, которая могла быть интерпретирована в цифровом виде. Формальная модель данных должна представлять собой множество структур данных, ограничений целостности и операций манипулирования данными. С помощью такой модели данных должны быть представлены объекты предметной области и взаимосвязи между ними.

Имея на руках такую модель данных, достаточно гибкую, но в то же время наиболее полно отображающая все данные и взаимосвязи между ними, можно легко решать содержательные аналитические задачи по ним. Находить количественные показатели. Строить графики и распределения по нетривиальным взаимосвязям, которые напрямую, используя лишь только первоначальные неструктурированные данные, построить очень трудно. Оперировать со структурированными данными помогают декларативные языки программирования.

После того как были получены численные результаты встаёт вопрос об их визуализации. Это также непростая задача, ведь она не описывается математически формально. И здесь нужен опыт и личные ощущения в том, как правильно, не потеряв самих результатов, наиболее красиво и наглядно можно визуализовать данные. Этому будет посвящена отдельная глава.

1.1 Содержательная постановка задачи

Ставится задача разработки и реализации информационно-аналитической системы в области наукометрии. Исходными данными для системы является библиографическая база системы web of science. Экспериментальные данные представлены в ненормализованном виде. Требуется определить все зависимости и провести нормализацию, опираясь на теоретический фундамент теории реляционных баз данных. Требуется разработать модель аналитического хранилища, реализовать процедуру загрузки данных в него. Провести анализ и визуализацию распределения публикаций по авторам, научным центрам, странам и годам. Разработать набор информационных панелей под аналитические задачи. Провести сетевое моделирование данных. Построить и визуализировать графа соавторства стран и ключевых слов в статьях.

Итак, рассматривается коллекция научных публикаций. Наша цель заключается в формализации имеющихся данных и преобразовании их в наиболее удобное представление для дальнейшего анализа. Далее необходимо загрузить данные в базу и максимально эффективно их визуализировать.

1.2 Основные понятия

Можно по-разному характеризовать понятие модели данных. С одной стороны, модель данных – это способ структурирования данных, которые рассматриваются как некоторая абстракция в отрыве от предметной области. С другой стороны, модель данных – это инструмент представления концептуальной модели предметной области и динамики ее изменения в виде базы данных.

Формальная модель данных — это абстрактное, самодостаточное, логическое определение *объектов* и *операторов*, в совокупности составляющих абстрактную машину доступа к данным. Объекты позволяют моделировать структуру данных, а операторы – поведение данных.

В классической модели данных есть формальная теория представления и обработки данных, которая включает, по меньшей мере, три аспекта:

- *Аспект структуры* : методы описания типов и логических структур данных;
- *Аспект манипуляции* : методы манипулирования данными;
- *Аспект целостности* : методы описания и поддержки целостности данных.

Аспект структуры определяет, что из себя логически представляет модель данных, аспект манипуляции определяет способы перехода между состояниями данных, аспект целостности определяет средства описаний корректных состояний данных. Учитывая обе вышеуказанные стороны, определим основные структуры моделей данных, используемые для представления концептуальной модели предметной области (сущностей, атрибутов, связей).

Поле – наименьшая поименованная единица данных. Используется для представления значения атрибута. *Запись* – поименованная совокупность полей. Используется для представления совокупности атрибутов сущности. *Экземпляр записи* – запись с конкретными значениями полей.

Важнейшим понятием концептуальной модели является понятие связи между *сущностями*. В моделях данных соответствующее понятие отражается понятием

групповое отношение, что есть поименованное бинарное отношение, заданное на двух множествах экземпляров рассматриваемых групп. По характеру бинарных связей различают групповые отношения вида $1:1$ (один к одному), $1:M$ (один ко многим), $M:N$ (многие ко многим). Пары чисел называют коэффициентами группового отношения. В групповом отношении один член группы назначается владельцем отношения, другой – членом отношения.

Для представления группового отношения используется две формы.

Графовая форма – группы изображаются вершинами графа, связи между группами – дугами, направленными от группы-владельца к группе-члену с указанием имени отношения и коэффициента.

По типу графов различают:

- *Иерархическую модель* – дерево;
- *Сетевую модель* – ориентированный граф общего вида.

Табличная форма – связь между группами изображается таблицей, столбцы которой представляют ключи соответствующих групп. Для формального описания таблицы используется математическое (теоретико-множественное) понятие отношения. Соответствующая модель данных называется *реляционной моделью*. Рассмотрим более подробно эту модель, так как она является определяющей в современном мире. Промышленность, коммерческие организации, правительства – все так или иначе используют в своей работе базы данных, основанные на реляционном принципе.

1.3 Способ построение модели данных

Формальная модель данных описывается следующим образом:

1. Определяются типы и характеристики логических структур данных (полей, записей, файлов);

2. Описываются правила составления структур более общего типа из структур более простых типов;

3. Описываются возможные действия над структурами и правила их выполнения, включающие:

   - Основные элементарные операции над данными;
   - Обобщенные операции (процедуры);
   - Средства контроля относительно простых условий корректности ввода данных (ограничения);
   - Средства контроля сколь угодно сложных условий корректности выполнения определенных действий (правила).

В качестве основных элементарных операций обычно рассматриваются следующие: поиск записи с заданным значением ключа, чтение нужной записи, добавление записи, корректировка, удаление. В моделях данных также предусматриваются специальные операции для установления групповых отношений. Обобщенные операции или процедуры – последовательность операций, реализующая определенный алгоритм обработки данных. Средства контроля используются для реализации ограничений целостности концептуальной модели. Простейшие средства контроля ограничения используются для реализации, как внешних ограничений концептуальной модели, так и внутренних ограничений модели данных. В качестве последних ограничений, в частности, реализованы ограничения на ввод данных несоответствующего типа, несоответствующей характеристики (по числу битов, по числу полей, по количеству записей). Более сложные средства контроля (правила) позволяют вызывать выполнение определенной последовательности операций (сколь угодно сложной) при изменении или добавлении данных в систему и тем самым реализовывать ограничения целостности, описанные с помощью специальных конструкций.

Глава 2

Реляционное моделирование

Реляционная модель данных – логическая модель данных, которая является приложением к задачам обработки данных таких разделов математики, как теория множеств и логика первого порядка.

Термин *реляционный* означает, что теория основана на понятии *отношения*, что есть математическая структура, которая формально определяет свойства различных объектов и их взаимосвязи. Например, отношение равенства, делимости, подобия, параллельности и другие. Наглядно теоретико-множественное отношение можно представить в виде таблицы, каждая строка которой содержит конкретные примеры объектов, связанных данным отношением. Отсюда в качестве неформального синонима термину *отношение* часто встречается слово *таблица* и можно сказать, что реляционная модель данных представляет информацию в виде совокупности связанных таблиц, которые называются отношениями или реляциями. Реляционная модель является удобной и наиболее привычной формой представления данных.

Для реляционных баз данных верен информационный принцип: всё информационное наполнение базы данных представлено одним и только одним способом а именно – явным заданием значений атрибутов в кортежах отношений; в частности, нет никаких указателей (адресов), связывающих одно значение с другим.

Реляционные базы данных могут быть описаны так называемыми *ER-диаграммами*. *ER-модель* (от англ. entity-relationship model, модель «сущность – связь») – модель

данных, позволяющая описывать концептуальные схемы предметной области. ER-модель используется при высокоуровневом (концептуальном) проектировании баз данных. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями.

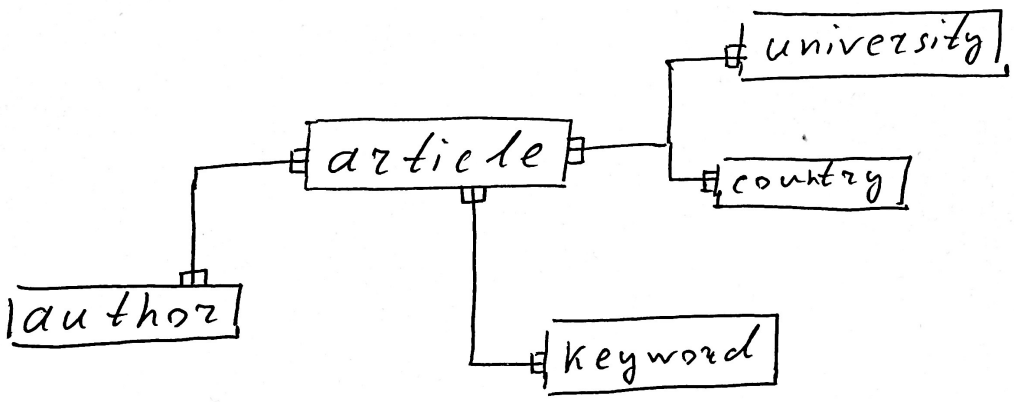


Рис. 2.1: ER-диаграмма «Научные статьи»

2.1 Исследование экспериментальных данных

Рассмотрим в качестве модельных данных коллекцию научных статей. Коллекция закодирована в файл SCI.TXT. Что мы имеем на начальном этапе работы: текст из 364222 строк по числу объектов-статей. Каждая строка – набор строк, которые заключены в кавычки и разделены запятыми. Наша цель – формализовать имеющиеся данные и преобразовать их в наиболее удобное представление для дальнейшего анализа.

2.2 Построение схемы отношения на базе одной таблицы

Следуя реляционному подходу составим таблицу (отношение) «Научные статьи». Записями в данной таблице будут строки исходного файла. Отдельными полями – фразы, заключённые в кавычки.

Для дальнейшей работы на потребуется «pandas» – программная библиотека на языке Python для обработки и анализа данных. Эта библиотека умеет работать со структурированными файлами как SCI.TXT, организацию которого можно отнести к «CSV» (от англ. Comma-Separated Values — значения, разделённые запятыми) – текстовый формат, предназначенный для представления табличных данных. Значения столбцов 3 и с 9 по 14 оставили целиком пустыми (со значением NaN), поэтому их не нужно загружать в память. Также нам известны имена заголовки столбцов (имена атрибутов). Тогда при выполнении соответствующего кода, получаем таблицу, фрагмент которой представлен (Таблица 2.1).

Таблица 2.1: Колекция научных статей

	Authors	Title	Source	Author keywords	...
0	Marchand-J Pigeon-M Bager-D Talbot-C	Influence of Chloride Solution Concentration o...	ACI MATERIALS JOURNAL 1999, Vol 96, Iss 4, pp ...	Deicers; Ice Formation; Porosity	...
1	Edvardsen-C	Water Permeability and Autogenous Healing of C...	ACI MATERIALS JOURNAL 1999, Vol 96, Iss 4, pp ...	Autogenous Healing; Concretes; Cracking (Fract...	...
2	Pauly-TM Lingvall-P	Effects of Mechanical Forage Treatment and Sur...	ACTA AGRICULTURAE SCANDINAVICA SECTION A-ANIMA...	Chopping; Clostridium spp.; Forage Wagon; Harv...	...
...	...	...	...	...	...

Все атрибуты отношения «Научные статьи»: Authors, Title, Source, Language, Document type, IDS/Book number, Number of related records, Number of references, RCs, Author keywords, Editor keywords, Abstract, References, – изначально относятся к одному домену – символьные строки. Это годится лишь для визуального представления данных, но никак не для работы с ними. Атрибут Source разобьём на два самостоятельных атрибута: journal и year первый из которых относится к домену «строка», а второй к домену «целое неотрицательное число». Атрибут RCs из строкового типа переопределим в домен «список пар строк» и переименуем university_country, первое слово в паре определяет университет, а второе – страну. Атрибуты Title, Language, Document type, Abstract оставим в домене «строка». Тогда получаем следующее множество доменов отношения «Научные статьи» с их атрибутами:

- Строка: title, language, doctype, abstract, journal
- Целое неотрицательное число: related records count, references count, year
- Список строк: authors
- Список пар строк: university_country, keywords

2.3 Анализ данных на базе одной многоструктурной таблицы

Теперь данные уже могут быть подвержены некоторому анализу. Так, например, можно составить словарь (список) всех авторов публикаций. Следующий алгоритм пройдётся по всем записям таблицы и соединит списки авторов по каждой отдельной статье в один список уникальных авторов. Например, список первых десяти авторов, упорядоченный по алфавиту (всего в коллекции 404865 авторов):

['A-J', 'A-K', 'AAgren-H', 'AElbasit-IE', 'AElgadir-TME', 'AErtebjerg-G', 'AH-T', 'AHearn-MF', 'AIZoughool-M', 'AKerblom-HK']

Аналогично составим словари стран, университетов. Словари авторских и редакторских слов. Всего уникальных университетов – 69093, а стран – 195. Редакторских слов – 292424, авторских – 335645 (что приближается к числу всех научных публикаций).

Вывод – коллекция научных статей собрана со всего мира, ведь всего признанных стран 197 штук. Число авторских слов заметно больше, чем редакторских – это объясняется тем, что разные авторы двух публикаций с большой вероятностью напишут об одном и том же разными словами. У редакторов же стандартизация серьёзнее, и один и тот же редактор проверяет множество публикаций – в статьях с общим смыслом он будет выделять одни и те же ключевые слова.

Используя библиотеки языка Python, такие как Matplotlib и NumPy, построим графики распределений количества публикаций, ссылающихся на данную публикацию и количества публикаций, связанных с данной публикацией:

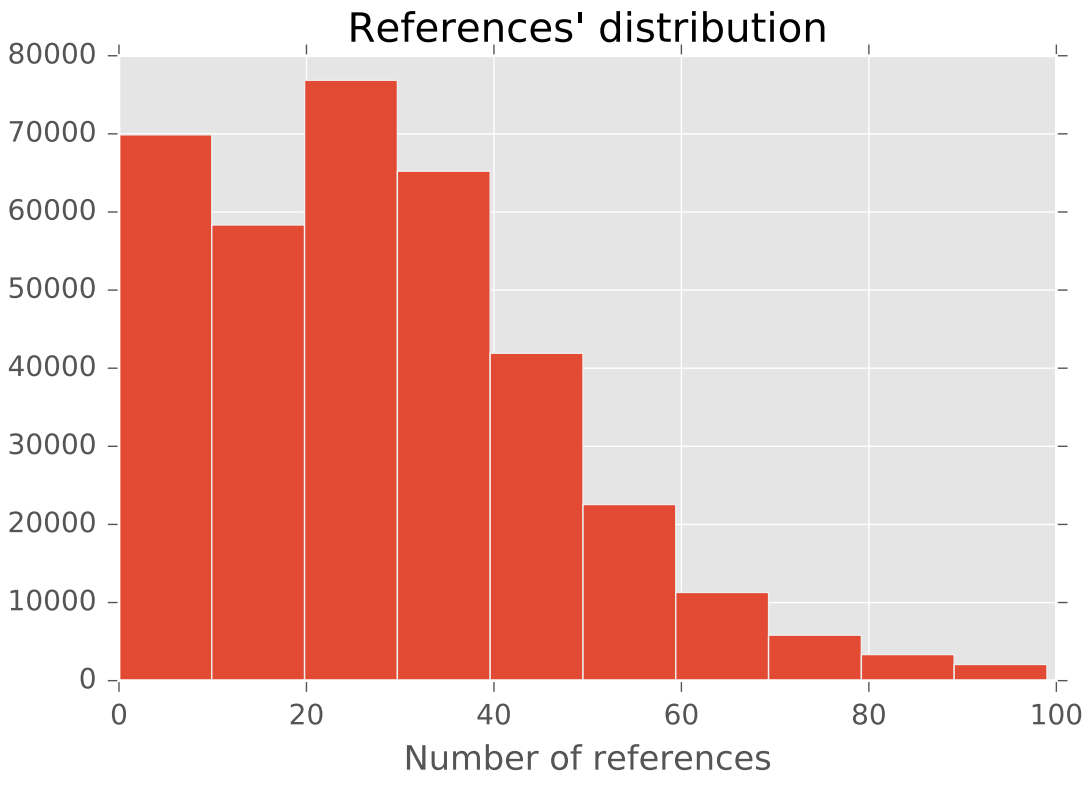


Рис. 2.2: Распределение числа ссылок в библиографии

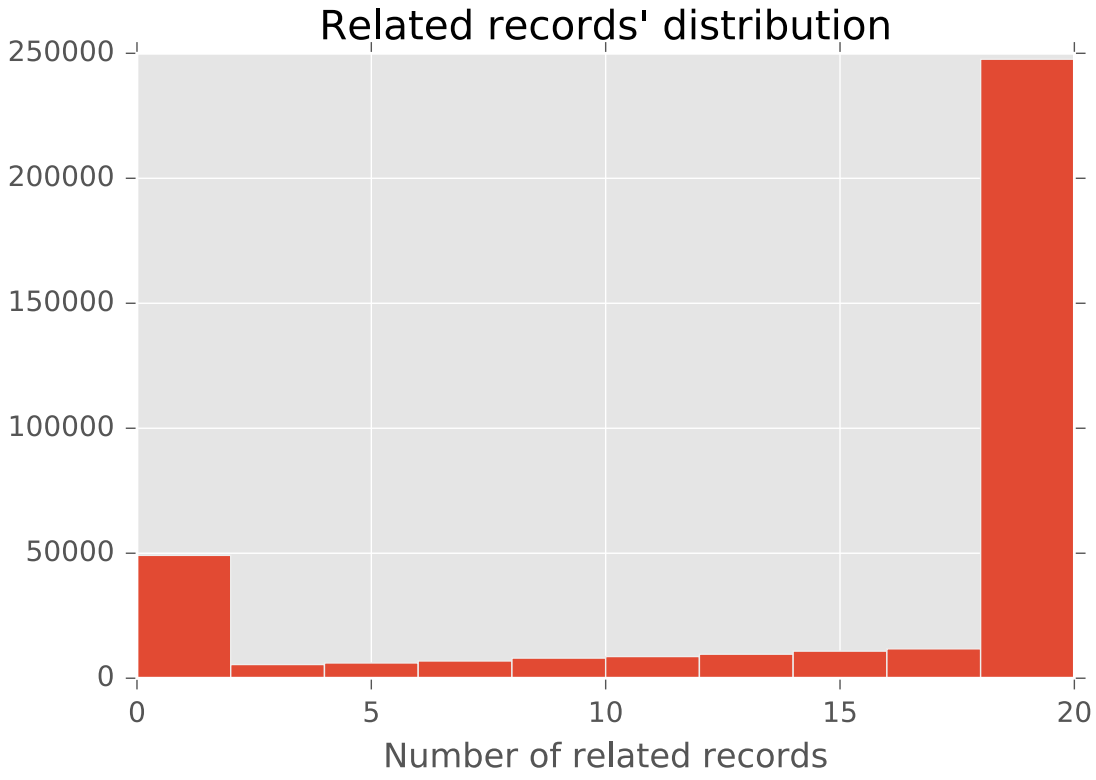


Рис. 2.3: Распределение числа родственных публикаций к данной

Число ссылок публикации в целом распределено равномерно от 0 до 40, и далее, это число уменьшается с квадратичной скоростью, и уже мало какие статьи ссылаются больше, чем на 80 источников (меньше 5 тысяч публикаций). Распределение числа родственных публикаций к данной имеет характерные пики в окрестности 0 (50 тысяч статей) и 20 (250 тысяч). Остальные 30 тысяч статей имеют распределили в целом равномерно число связанных публикаций от 0 до 20.

Отсюда можно заключить, что обычно в публикации указывают от 0 до 40 источников, достаточно большое число статей без (или практически без) источников (70 тысяч). Наверное, это совсем новые и оригинальные области исследований. 40 источников указывают уже сильно меньшее число статей (40 тысяч). А 60 – уже 10 тысяч.

Исходя из вида распределения связанных статей можно предположить, что есть области знаний хорошо освоенные, в которых написано большое число родственных

статей (250 тысяч), а есть новые и оригинальные области, в которых практически нет родственных публикаций (50 тысяч).

Проводить дальнейший анализ уже сложнее. Необходимо упрощать формальную модель, усложняя её математику. Сделаем переход от одной многоструктурной таблицы к системе связанных элементарных таблиц.

2.4 Нормализация базы данных

Проведём нормализацию коллекции научных статей. Изначально данные находятся в так называемой нулевой начальной форме (Рис. 2.4). В данных есть неатомарные атрибуты: множество авторов, ключевых слов, научных центров. Анализ данных в такой нормальной форме затруднителен. Первая задача – это привести данные к самой первой нормальной форме, когда каждый атрибут атомарен и не представляет собой множества.

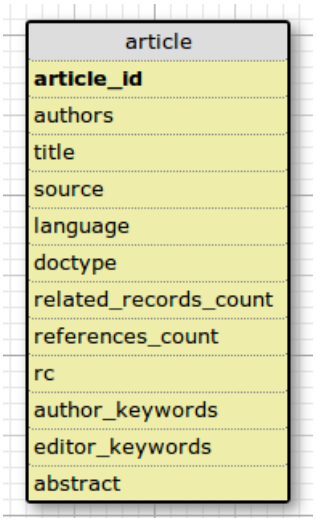


Рис. 2.4: База данных в нулевой нормальной форме

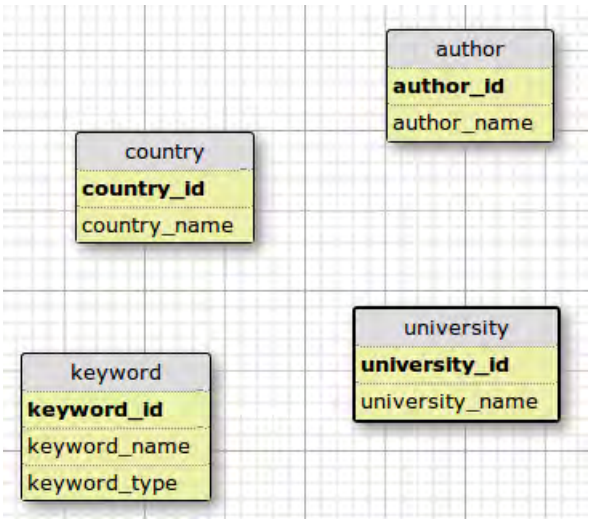


Рис. 2.5: Словари авторов, стран, университетов и ключевых слов

Для того чтобы привести данные к первой нормальной форме необходимо атрибуты, представляющие собой множества значений, выделить в отдельные сущности.

Построим словари авторов, стран и университетов. Также объединим авторские и редакторские ключевые слова в один словарь с полем типа ключевого слова (Рис. 2.5).

После того как новые сущности были выделены в свои таблицы, необходимо связать их в единую диаграмму. Связи типа "многие-ко-многим"моделируются при помощи развязочных таблиц. Тогда напишем сценарий, строящий развязочные таблицы, представленные на (Рис. 2.7). Стоит отметить, что есть развязочная таблица, содержащая в себе три поля, так как все три сущности: статьи, университеты и страны связаны друг с другом.

Декомпозиция одной сущности на несколько связанных сущностей без потери информации обеспечивается теоремой {1} *Хита*. Она утверждает, что если при использовании операции соединения JOIN для двух разделённых таблиц, получается первоначальная таблица (до разделения), то такая декомпозиция происходит без потери информации.

Теорема 1. *Пусть дано отношение* $r(A, B, C)$ *Если* r *удовлетворяет функциональной зависимости* $A \to B$*, то оно равно соединению его проекций* $r[A, B]$ *и* $r[A, C]$

$(A \to B) \Rightarrow (r(A, B, C) = r[A, B]\ JOIN\ r[A, C])$

Это действительно так в нашем случае, и после соединения отношений статей, авторов и их развязочной таблицы, получится одна таблица, каждый кортеж которой будет представлять собой одного автора и связанной с ним одной публикации. И так для любого автора и любой публикации.

Дальше необходимо выделить в отдельные сущности множества журналов, языков и типов документов. Так как здесь могут возникнуть аномалии обновления – невозможно добавить новый журнал, когда в нём ещё нет ни одной публикации. Равно как и при удалении всех публикаций с данным журналом, пропадает информация о самом журнале. Тогда выделим словари для этих сущностей. Развязочные таблицы в данном случае составлять не нужно, так как имеет место связь *многие к одному* в отношении публикаций к журналам, языкам и типам.

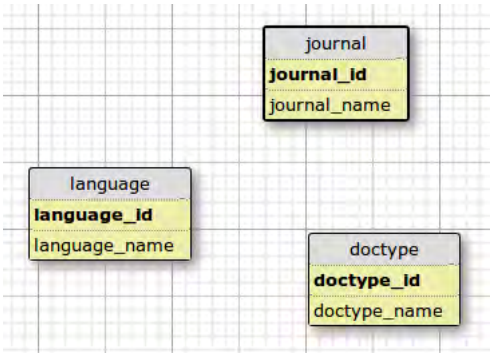


Рис. 2.6: Словари языка, изданий и типов статей

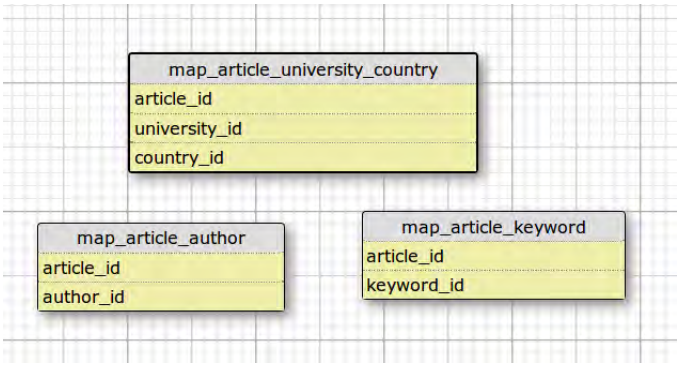


Рис. 2.7: Развязочные таблицы

Итак, схема базы данных «Научные публикации» в третьей нормальной форме представлена следующей схемой {2.8}. Все исходные данные приведены к данному представлению. Был написан сценарий, строящий по исходной таблице 11 нормализованных логически связанных таблиц.

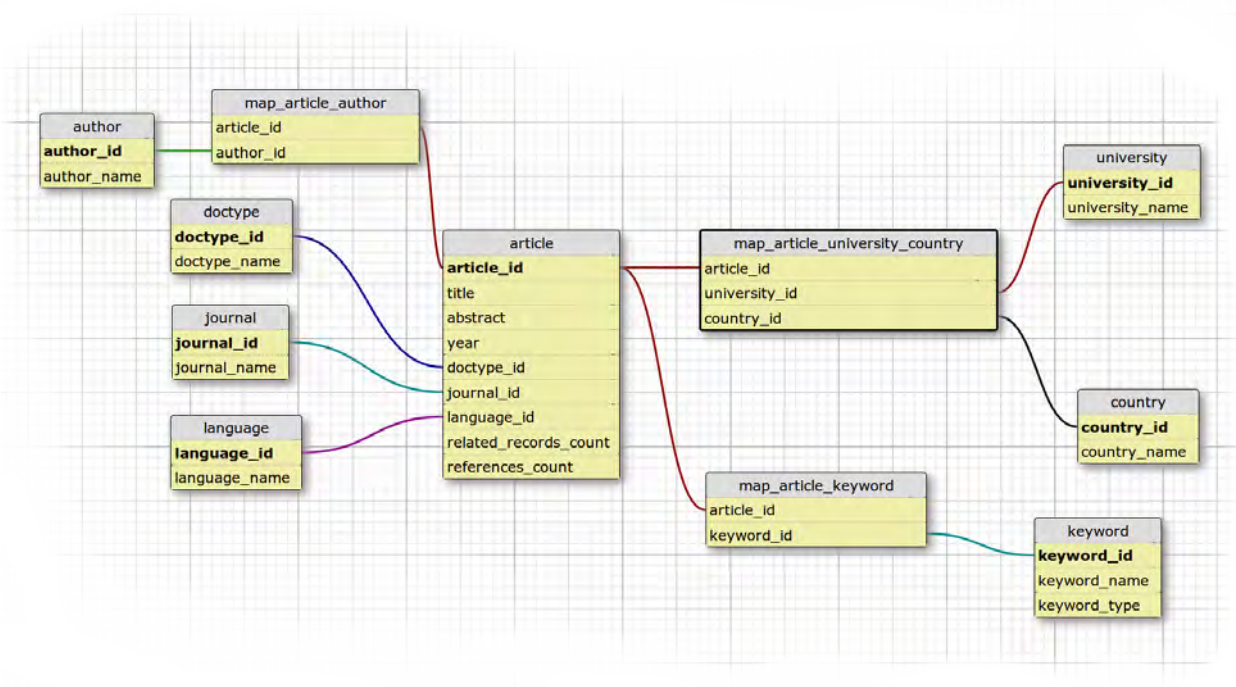


Рис. 2.8: Схема базы данных «Научные статьи»

2.5 Алгоритмы

Data: L – список множеств авторов по статьям

[множество_авторов] – элемент L

$M \leftarrow list()$;

for $l \in L$ **do**

  $M.extend(l)$

end

$M \leftarrow sorted(M)$;

$A \leftarrow dict()$;

for $author\_number, author\_name \in enumerate(M)$ **do**

  $A.setdefault(author\_name, author\_number)$

end

Result: A – словарь всех авторов

{имя_автора: номер_автора} – элемент A

Algorithm 1: Создание словаря авторов

Data: L – список множеств авторов по статьям

[множество_авторов] – элемент L

Data: D – словарь всех авторов

{имя_автора: номер_автора} – элемент D

$N \leftarrow list()$;

for $article\_number, author\_set \in enumerate(L)$ **do**

  for $author \in author\_set$ **do**

    $N.append((D[author], article\_number))$

  end

end

Result: N – развязочная таблица статей-авторов

[имя_автора, номер_статьи] – элемент N

Algorithm 2: Создание развязочной таблицы по статьям-авторам

Глава 3

Бизнес-аналитика

Информационные системы серьезного предприятия, как правило, содержат приложения, предназначенные для комплексного анализа данных, их динамики, тенденций. Соответственно, основными потребителями результатов анализа становится топ-менеджмент. Такой анализ, в конечном итоге, призван содействовать принятию решений. А чтобы принять любое управленческое решение необходимо обладать необходимой для этого информацией, обычно количественной. Для этого необходимо эти данные собрать из всех информационных систем предприятия, привести к общему формату и уже потом анализировать. Для этого создают хранилища данных.

Бизнес-интеллект (Buisness Intelligence) – это всеохватывающий термин, предложенный Говардом Дренсером в 1989 году для описания всевозможных концепций и методов, повышающих эффективность бизнеса путём использования систем анализа данных, которые позволяют быстро принимать решения. Средства BI интегрируют данные из OLTP-систем и трансформируют их в сведения о текущем состоянии и динамике бизнеса в целом. Можно сказать, что BI – это инструменты и приложения для поиска, анализа, моделирования и доставки информации, необходимой для принятия решений.

Была поставлена задача провести исследование данных в области наукометрии в рамках методологии BI.

Рассмотрим парадигму ETL, как один из основных процессов в управлении хранилищами данных, который включает в себя:

- Извлечение данных из внешних источников;
- Их трансформация и очистка;
- И загрузка их в хранилище данных.

Первый пункт получения данных был выполнен в процессе получения файла от научного руководителя. То есть он в данном случае выступал в роли внешнего источника. Второй пункт – трансформация и очистка. Был решён в предущей главе посредством нормализации данных. Третий пункт и дальнейшую работу с хранилищем рассмотрим отдельно.

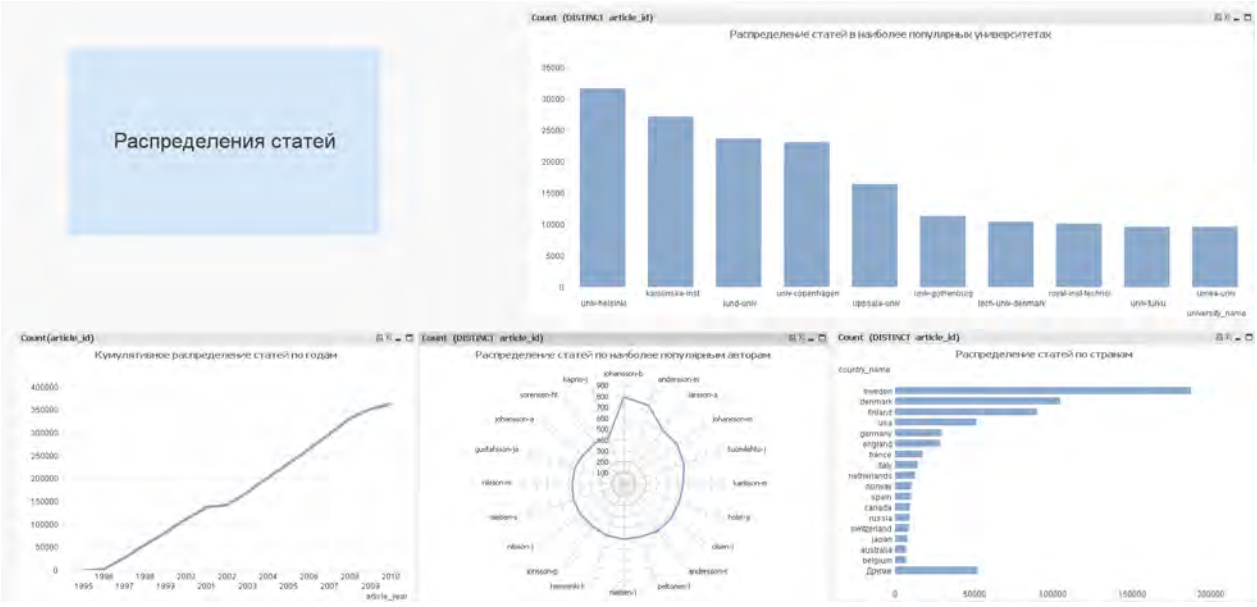


Рис. 3.1: Информационная панель QlikView

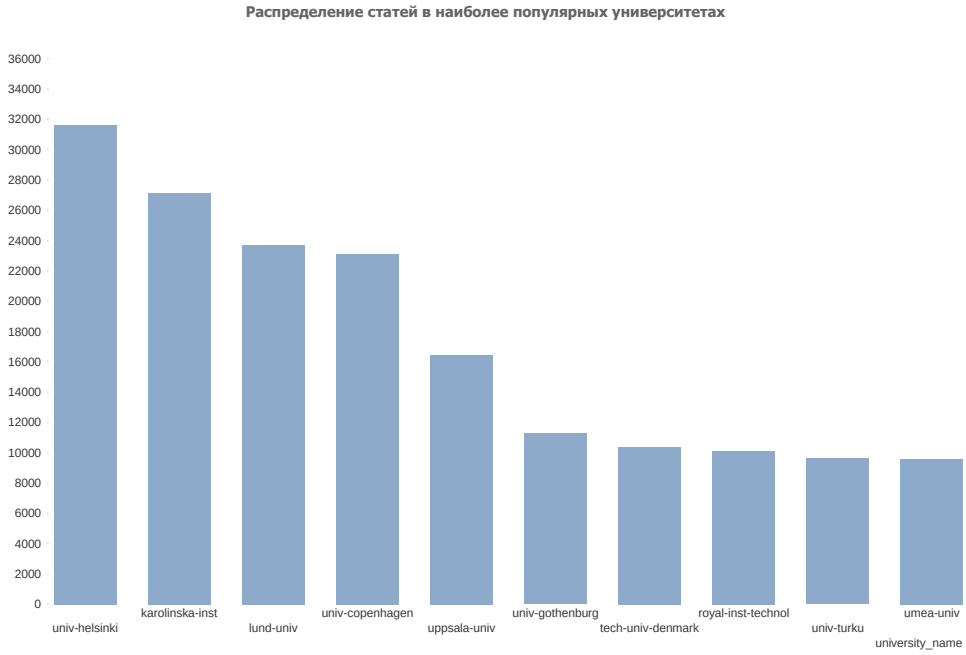
3.1 Загрузка данных и разработка информационных панелей

Третий пункт – загрузка данных. Налажена работа с серверной БД. Как основная система была выбрана свободная объектно-реляционная система управления базами данных PostgreSQL. Написан соответствующий сценарий DDL. Данные загружены на сервер PostgreSQL. Далее, построена витрина данных в системе QlikView {3.1}. В ней решены следующие задачи.

3.2 Решения аналитических задач и визуализация

Поставим и решим задачи на нормализованных данных. Оперативный анализ количества публикаций. В данной задаче выбранные для анализа объекты рассматриваются индивидуально. Категории одного измерения рассматриваются независимо. Категории разных измерений ограничивают количество публикаций. Для условия используются измерения: авторы, научные центры, страны, годы.

Необходимо рассчитать распределение статей в наиболее популярных университетах, кумулятивное распределение статей по годам, распределение статей по странам и распределение статей по наиболее популярным авторам. Решения аналитических задач представлены следующими графиками {3.2} и {3.3}.



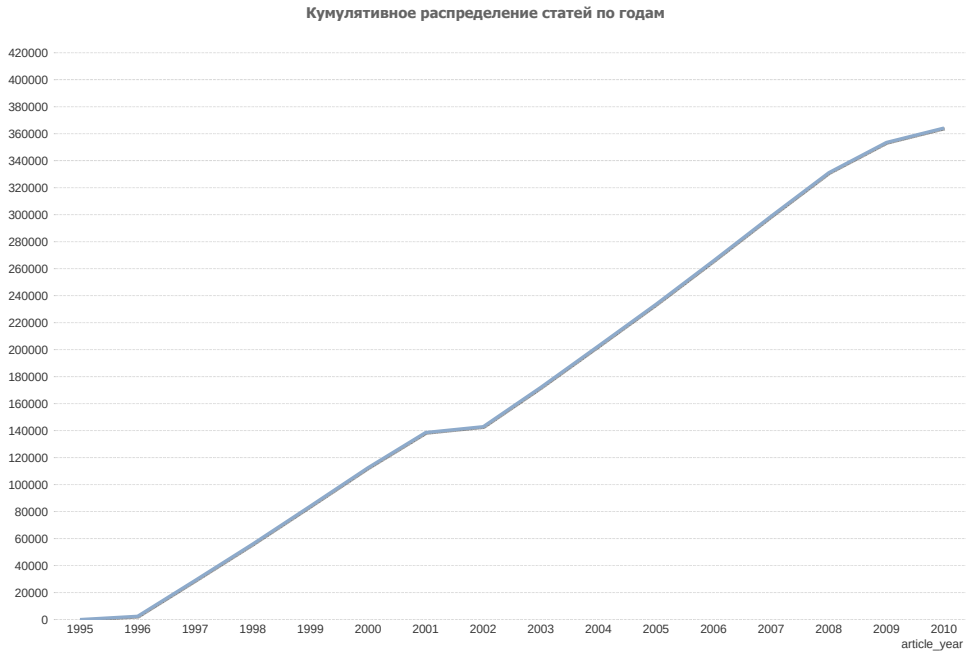
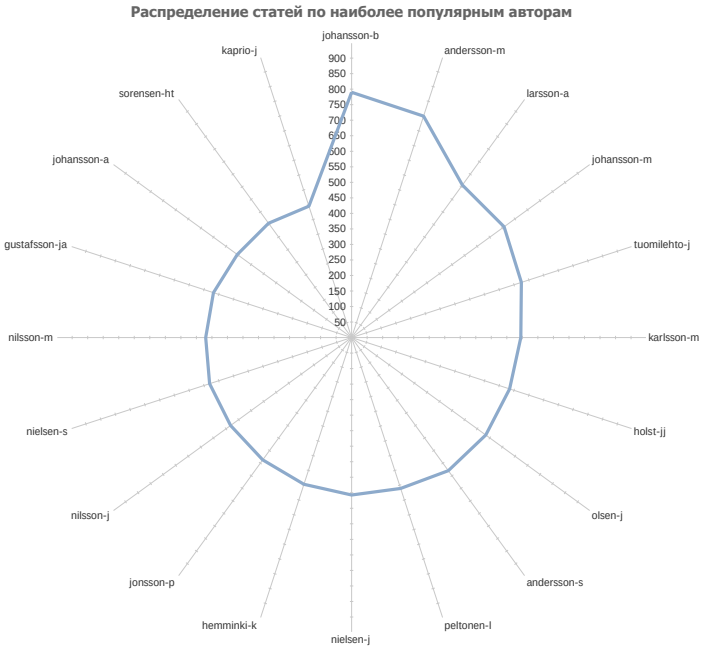


Рис. 3.2: Распределения статей по университетам и годам



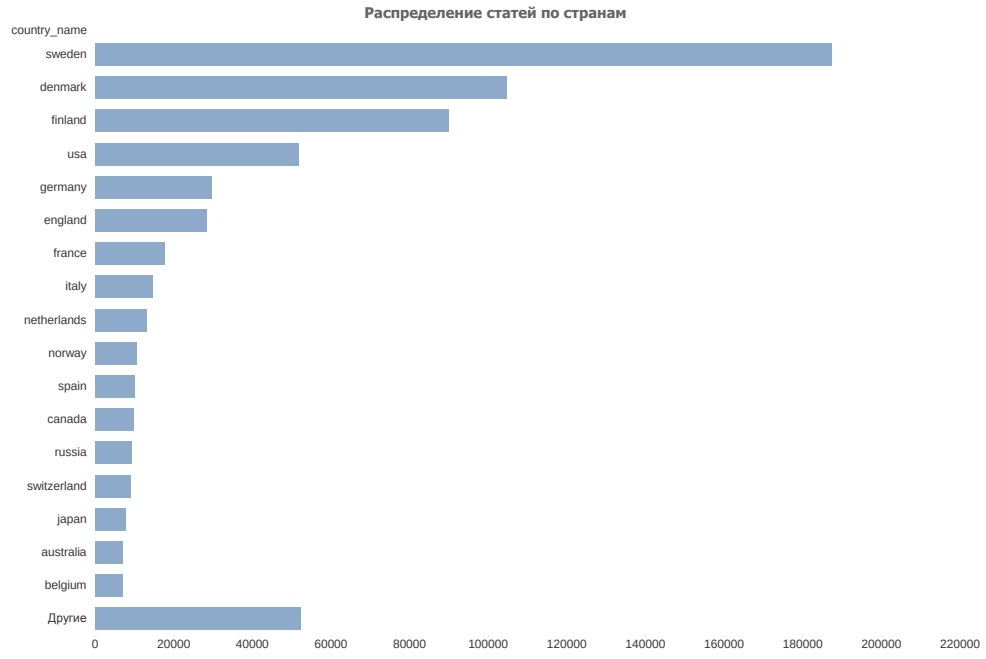


Рис. 3.3: Распределения статей по авторам и странам

Глава 4

Стохастические блоковые модели [6] и граф соавторства

Необходимо оценить степень научного сотрудничества между странами. Тогда соответствующая математическа постановка: необходимо построить граф соавторства по странам и ключевым словам. Сначала решим задачу для стран. Рассмотрим мультиграф $G = (V, E)$, где V – множество стран. Будем говорить, что страны v_i и v_j связаны ребром e_{ij}, если $\exists$ публикация a, такая что $v_i, v_j \in a$. Количество совместных публикаций опредляет количество рёбер между странами. Для нормализованных данных эта задача эффективно решается реляционными операциями *group by* и *join* над развязочной таблицей между публикациями и странами. Тогда имеем список смежности графа G.

Общий подход к анализу графов состоит в описании их структуры через подграфы или *сообщества*, разбивая множества вершин V на подмножества b. Таким образом, стоит задача кластеризации графа. Выделение структуры сообществ необходимо для эффективной визуализации и принятия решений по графу. Рассматривается стохастический блоковый алгоритм кластеризации на основе генеративной модели.

4.1 Байесовский вывод структуры графа

Принципиальный подход к решению задачи кластеризации состоит в построении *генеративной модели*, содержащей в себе идею модульной структуры графа. Тогда структура может быть найдена посредством *вывода* параметров модели по данным. Более чётко, пусть есть разбиение $b = \{b_i\}$ графа по B группам, где $b_i \in [1, B]$ – группа вершины i. Тогда определена генеративная модель, описывающая граф G:

$$p(G, b, \theta)$$

где θ – параметры модели. Наблюдаемое *правдоподобие* описывается разбиением b и параметрами θ как $p(G|b, \theta)$. Также необходимо определить *априорную* вероятность разбиения и параметров: $p(b, \theta)$. *Обоснованность* модели определяется вероятностью

$$p(G) = \sum_{\theta, b} p(G|b, \theta)p(b, \theta)$$

.

Теперь всё готово, можно осуществлять вывод *апостериорной* вероятности посредством *формулы Байеса*:

$$p(b|G) = \frac{p(G, b)}{p(G)} = \frac{\sum\limits_{\theta} p(G, b, \theta)}{p(G)} = \frac{\sum\limits_{\theta} p(G|b, \theta)p(b, \theta)}{p(G)}$$

Во многих моделях, в том числе в описываемой ниже, параметры θ жёстко зафиксированы методом максимального правдоподобия $p(G|b, \hat{\theta}) = \max\limits_{\theta} p(G|b, \theta)$. Это позволяет существенно упростить модель к виду:

$$p(b|G) = \frac{p(G|b, \theta)p(b, \theta)}{p(G)}$$

которая может быть переписана в терминах *энергии*:

$$p(b|G) = \frac{\exp(-\Sigma)}{p(G)}$$

тогда

$$p(b|G) \to \max \Leftrightarrow \Sigma \to \min$$

$$\Sigma = -\ln p(G|\theta, b) - \ln p(\theta, b) \to \min_{\theta, b}$$

и найденные $\hat{b}$ определяют оптимальное разбиение графа. Найденное апостериорное распределение $p(b|G)$ позволяет естественным образом решить задачу о разбиении графа на пересекающиеся группы: для $\forall$ вершины i графа G определена вероятность $p(b_i = k|G)$ принадлежности этой вершины группе k. И $\hat{b}_i = \arg\max_k p(b_i = k|G)$.

Стохастическая блоковая модель вероятно простейший генеративный процесс, базирующийся на группах вершин. Матрица смежности $\mathbb{R}^{BxB}$ разбиения b содержит элементы e_{rs}, соответствующие числу рёбер между группами r и s. Вершины графа в этой модели будут принадлежать к одной группе b_i, обладая схожей вероятностью быть соединёнными с остальными вершинами графа. Процесс генераци графа по заданным параметрам представлен следующими рисунками {4.1} и {4.2}.

В классической модели предполагается, что рёбра расположены случайно равномерно внутри каждой группы, и вершины, входящие в одну группу, обладают схожими степенями. Откуда следует, что классическая модель достаточно бедна для большинства гетерогенных сетей. Улучшенная модель учитывает степени вершин при помощи вектора скрытых переменных $k = \{k_i\}$.

Вычислительная сложность алгоритма пропорциональна числу вершин и равна $O(V \ln^2 V)$. Вывод осуществляется посредством сэмплирования из марковских цепей MCMC. Также используется факторизованное представление апостериорной вероятность $p(b|G) = q(b) = \prod_j q(b_j)$.

Можно пойти дальше и построить вложенную стохастическую блоковую модель. Она состоит из L слоём обычной СБМ, причём каждый следующий слой G_2 задаёт

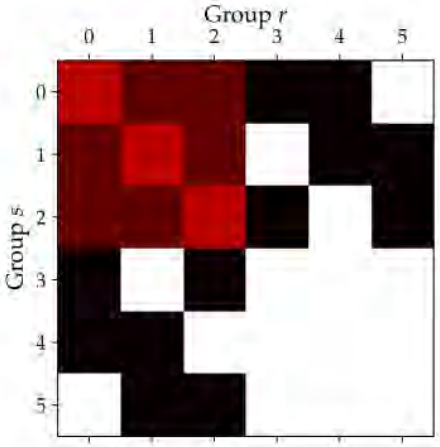


Рис. 4.1: Матрица смежности подсетей, определяющая число рёбер между кластерами

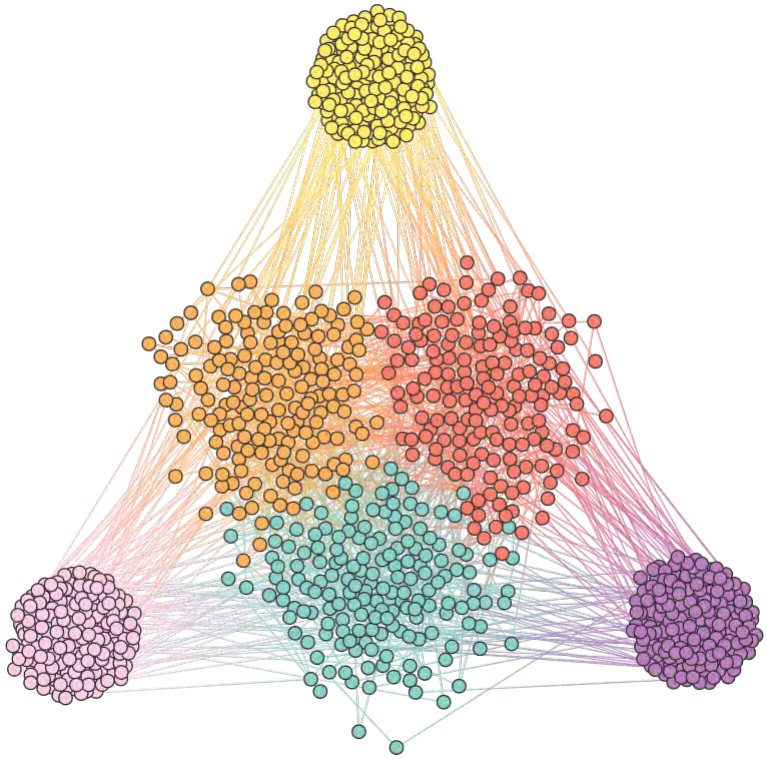


Рис. 4.2: Сгенерированння сеть

свою априорную вероятность $p(b, \hat{\theta}) = p(b|G_1)$ как апостериорную с предыдущего слоя. Пример вложенной модели представлен рисунком {4.3}.

4.2 Визуализация графа соавторaства стран

Вернёмся к исходной задаче. Необходимо было кластеризовать и визуализировать страны по их научному сотрудничеству. Для эффективной визуализации из исходных данных была выделена случайная подвыборка размерности 100к. Результаты визуализации представлены рисунками {4.4, 4.5, 4.6, 4.7}. Здесь мы видим, что страны объединяются в кластеры преимущественно по географической близости. Причём сильные страны образуют одноэлементные кластеры. Тогда как более слабые объединяются в группы. В силовой раскладке графа удалённость стран от центра определяет их вовлечённость в мировое научное сообщество.

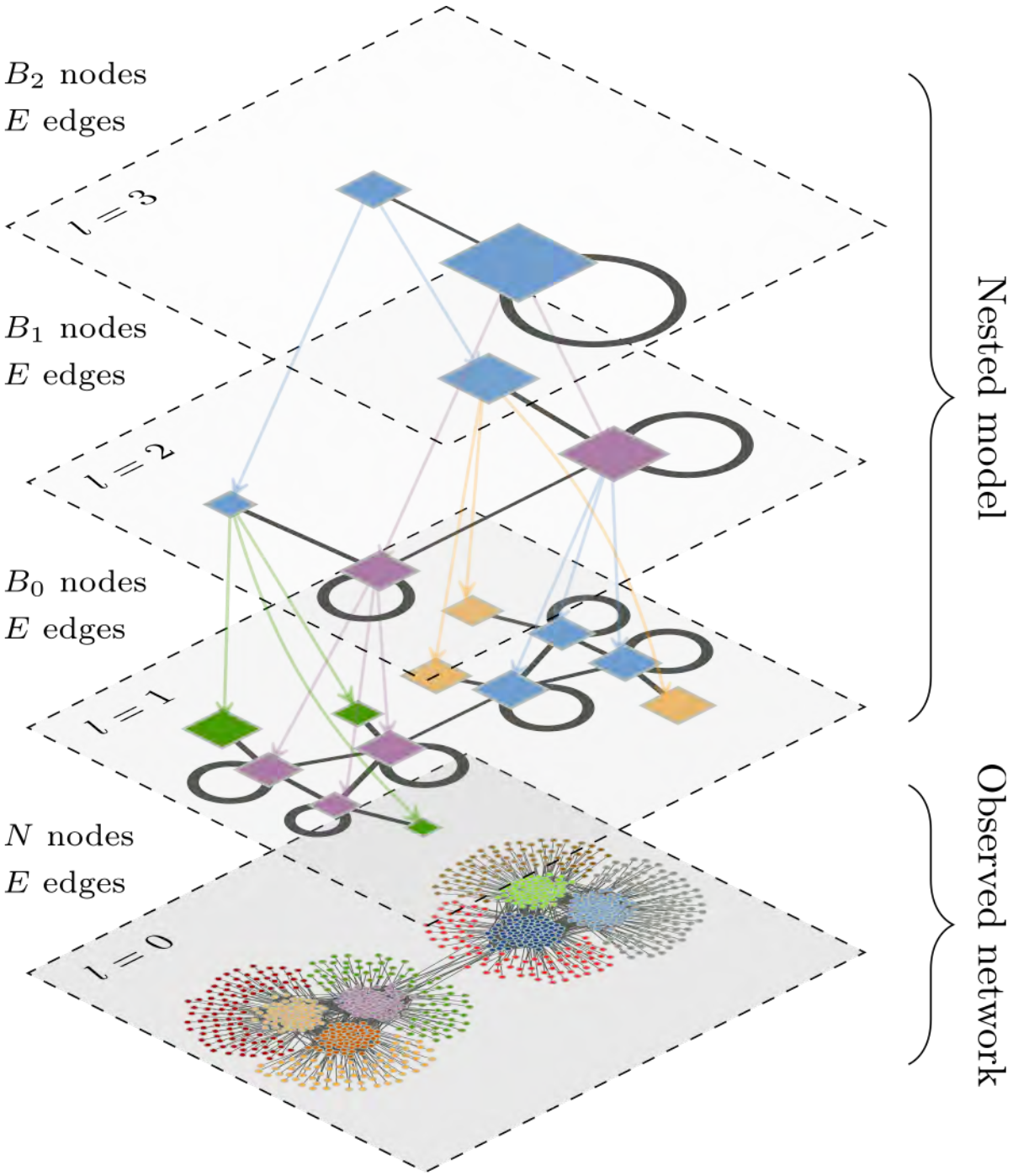


Рис. 4.3: Вложенная трёхуровневая стохастическая блоковая модель. Рассчитанное апостериорное распределение параметров предыдущего уровня переходит в априорное следующего.

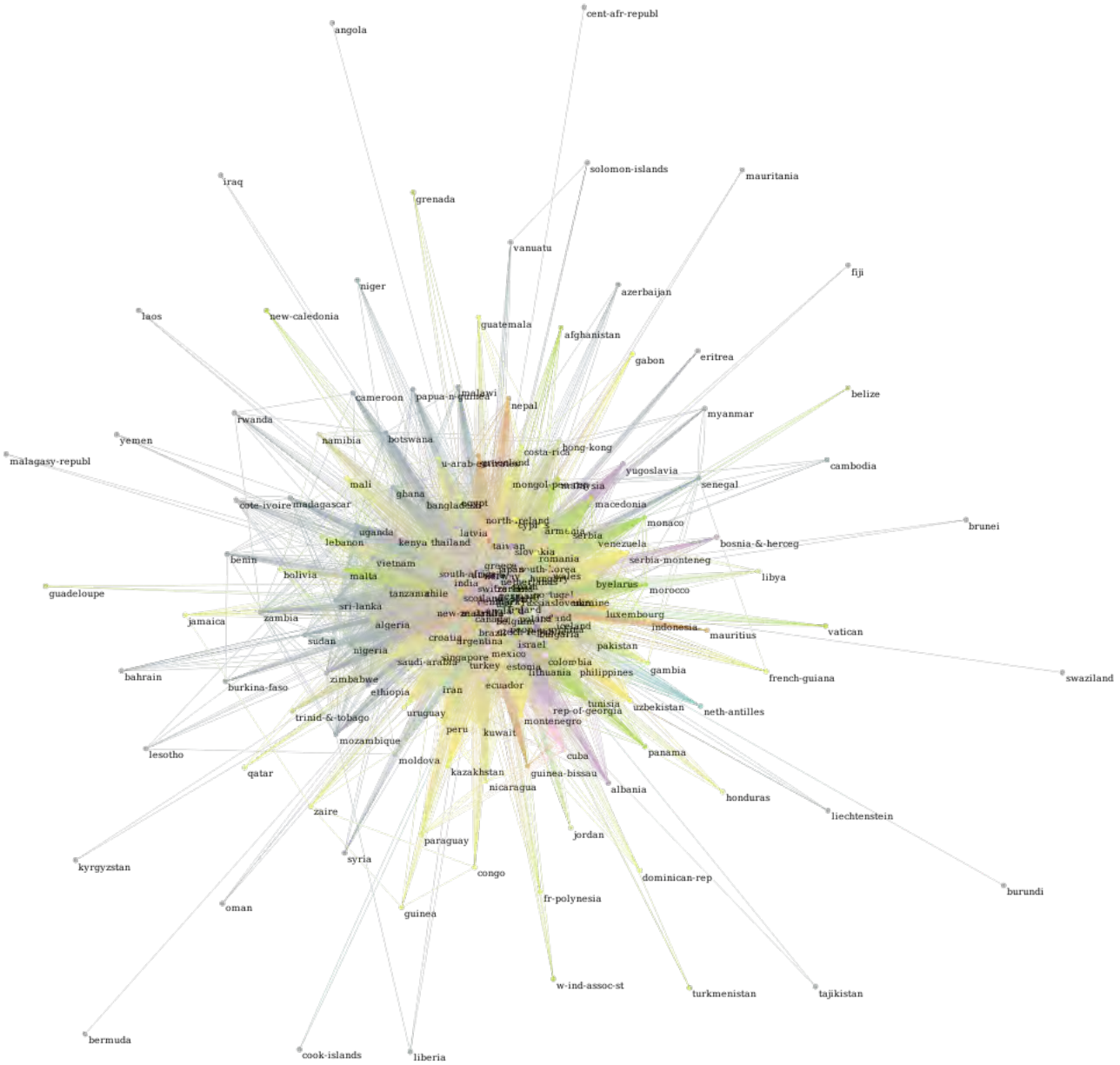


Рис. 4.4: Кластеризованный граф соавторства странам в силовой раскладке. Удалённость страны от центра определяет её вовлечённость в мировое научное сообщество.

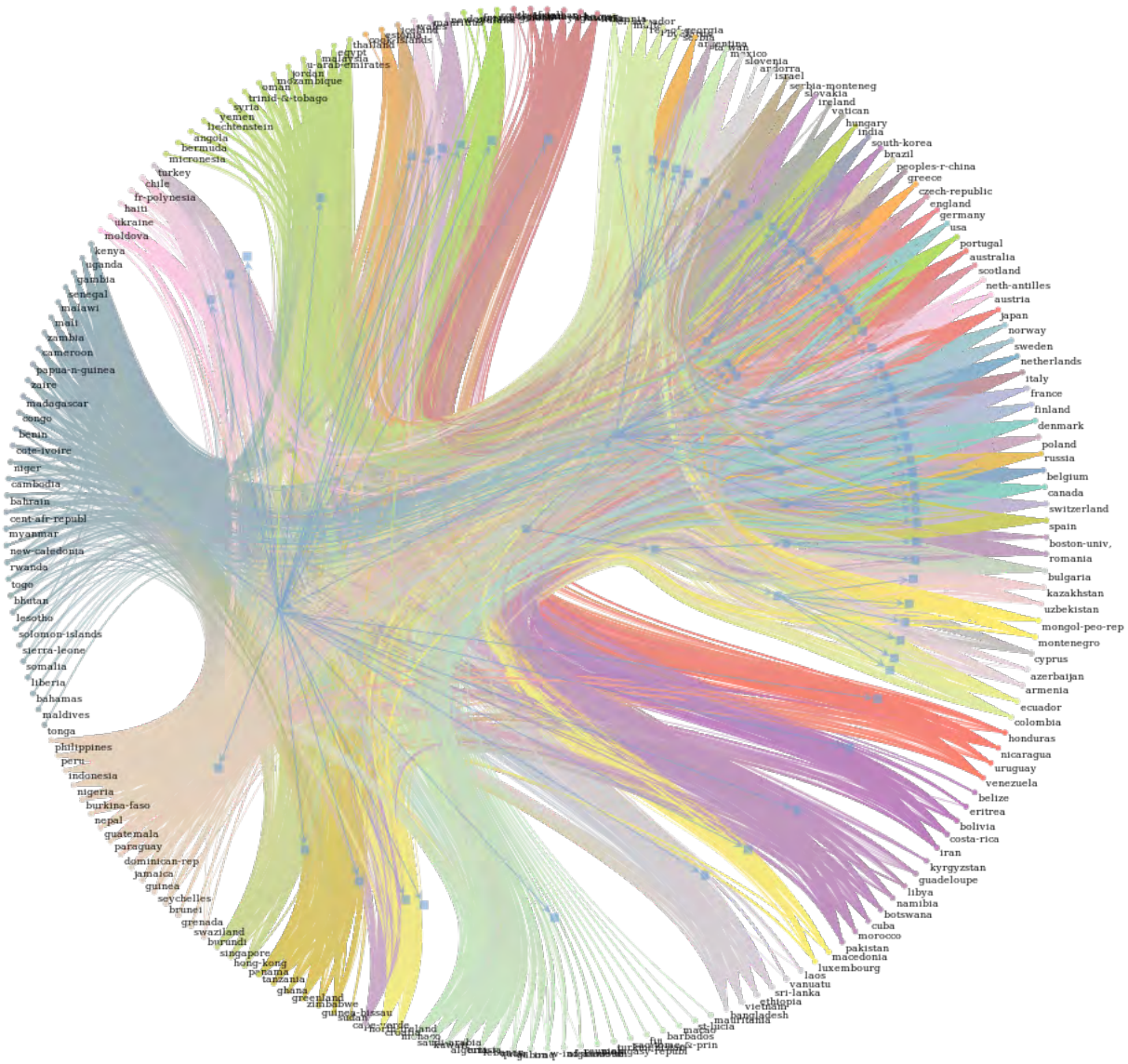


Рис. 4.5: Кластеризованный граф соавторства странам в круговой раскладке. Вложенная модель. Страны объединяются в кластеры преимущественно по географической близости. Причём сильные страны образуют одноэлементные кластеры. Тогда как более слабые объединяются в группы. Дерево синих квадратиков показывает объединения кластеров по уровням вложенной модели. Что есть иерархическая кластеризация.

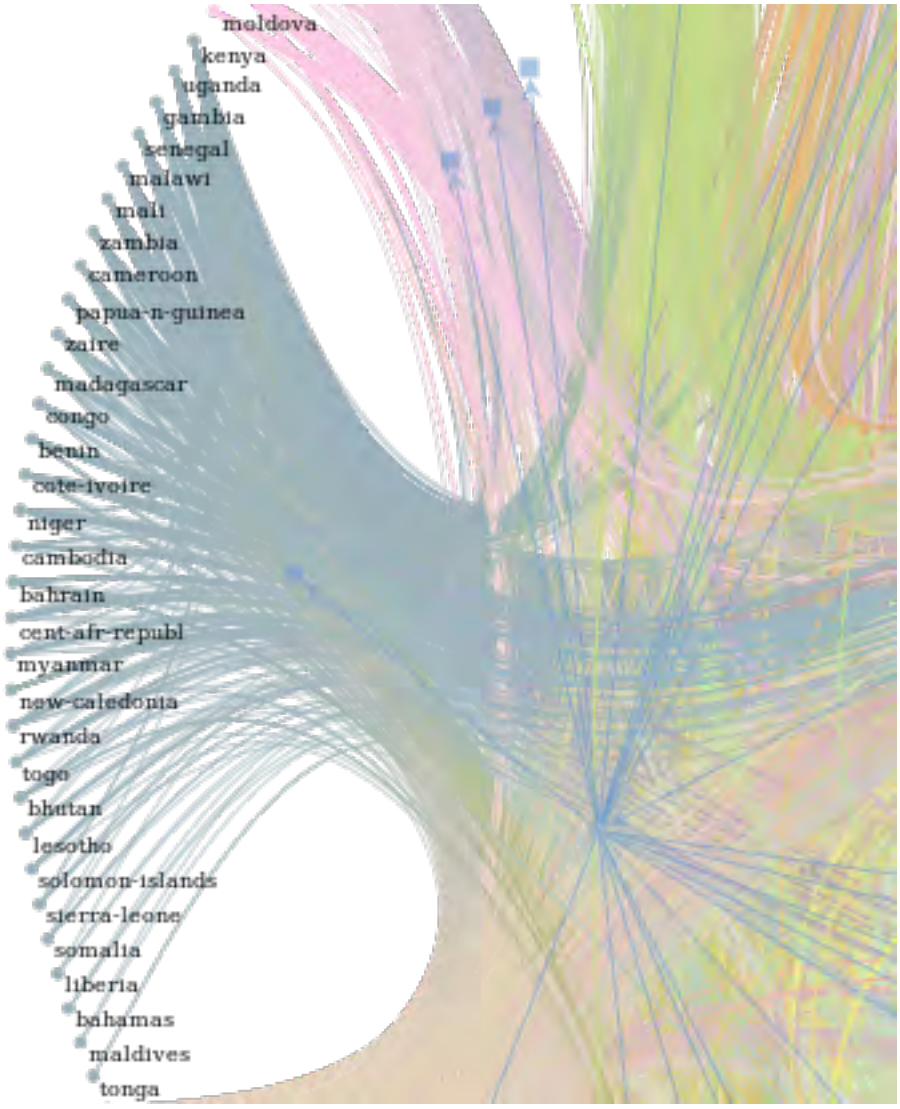


Рис. 4.6: Кластеризованный граф соавторства странам. Вложенная модель. Масштаб: выделена большая группа преимущественно африканских стран.

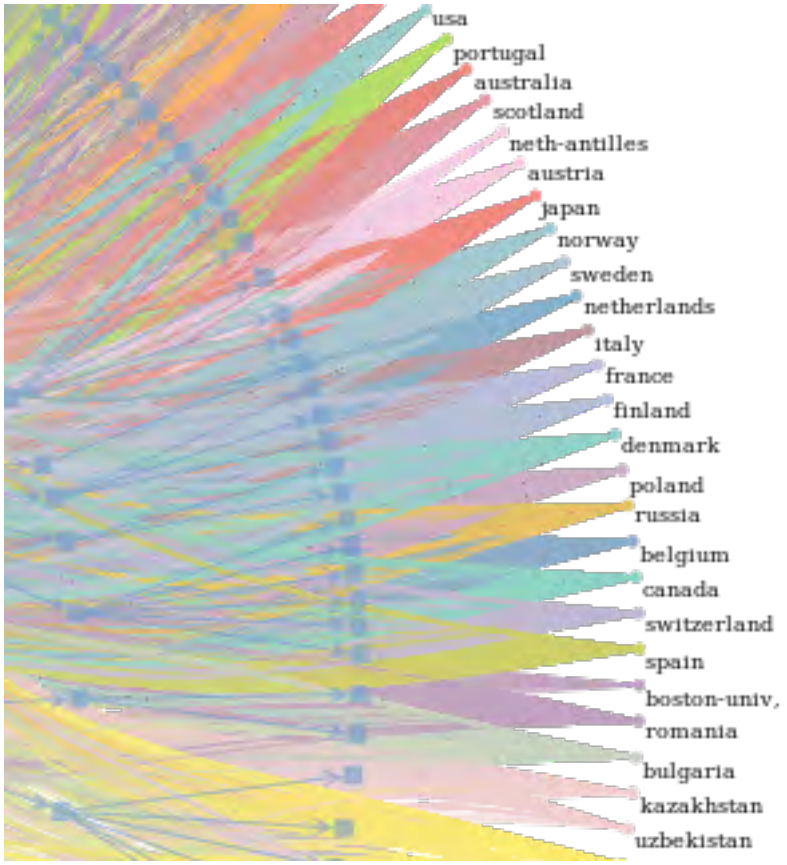


Рис. 4.7: Кластеризованный граф соавторства странам. Вложенная модель. Масштаб. Сильные страны образуют свои одноэлементные кластеры. Однако на предыдущих слоях иерархии объединяются в небольшие группы.

4.3 Раскраска карты мира

Наглядно отобразим научное взаимодействие между странами. Раскрасим карту мира таким образом, чтобы каждая страна была окрашена в соответствующей своей группе цвет. Попавшие в один кластер страны пишут много статей друг с другом.

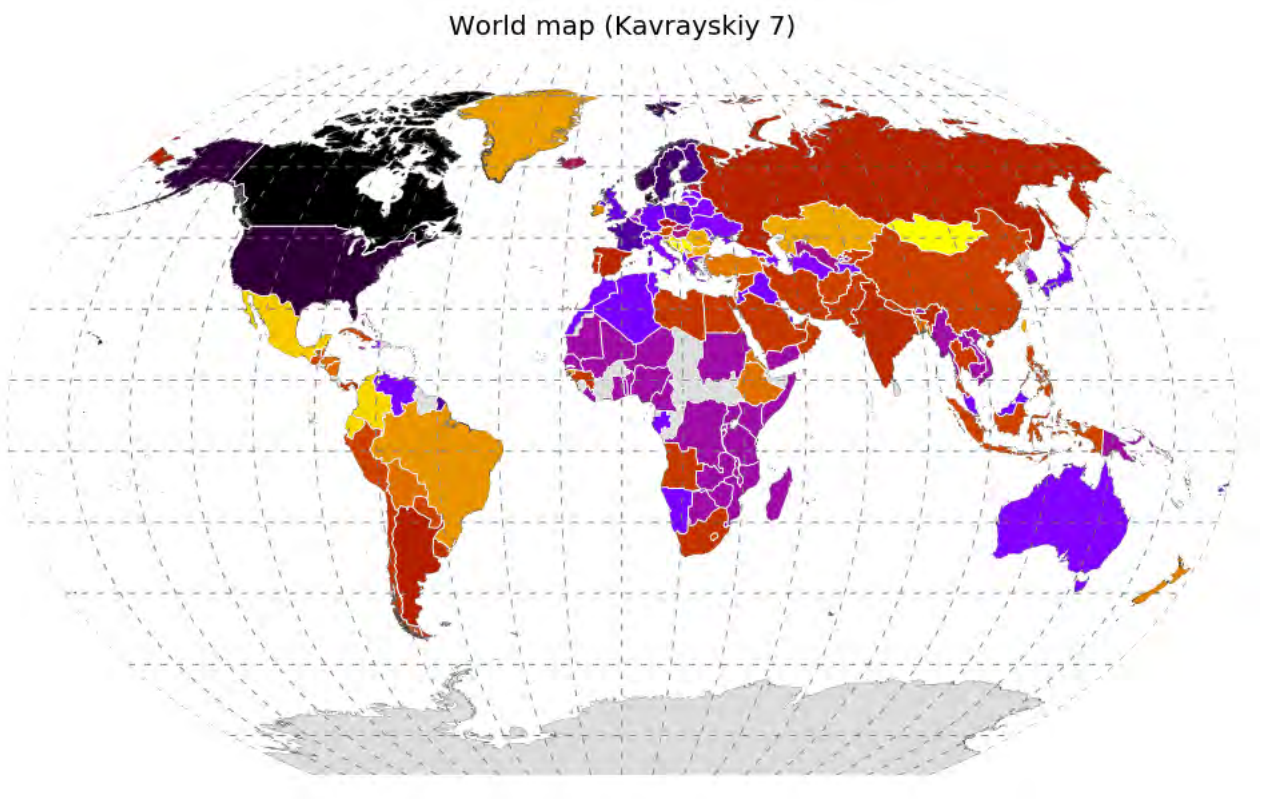


Рис. 4.8: Кластеризация стран по степени их научного взаимодействия

4.4 Визуализация графа соавторства ключевых слов

Решим аналогичную задачу для ключевых слов. Для эффективной визуализации из исходных данных была выделена случайная подвыборка статей размера 10к элементов. Взят подграф с вершинами степени больше десяти и меньше ста. Результаты визуализации представлены рисунками {4.10, 4.11, 4.9}.

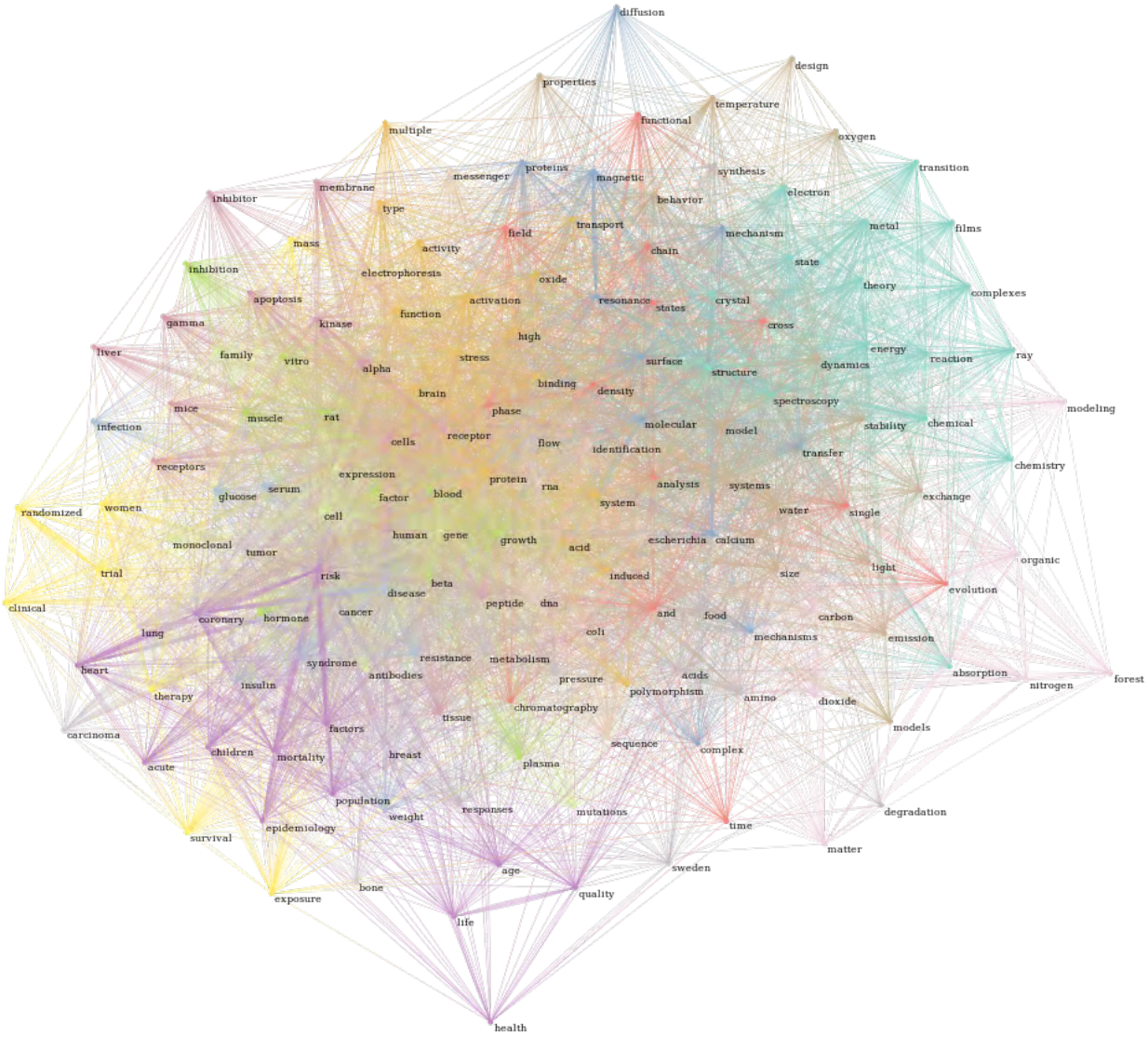


Рис. 4.9: Кластеризация ключевых слов в силовой раскладке. В отличие от стран в такой же раскладке, слова не образуют явную звездовидную структуру. Все вершины обладают схожими степенями. Это говорит о том, что в выделенной подвыборке ключевые слова равноценны в смысле их совместного упоминания в статьях. Жёлтый кластер, конечно, интересный получился: randomized, trial, clinical...women.

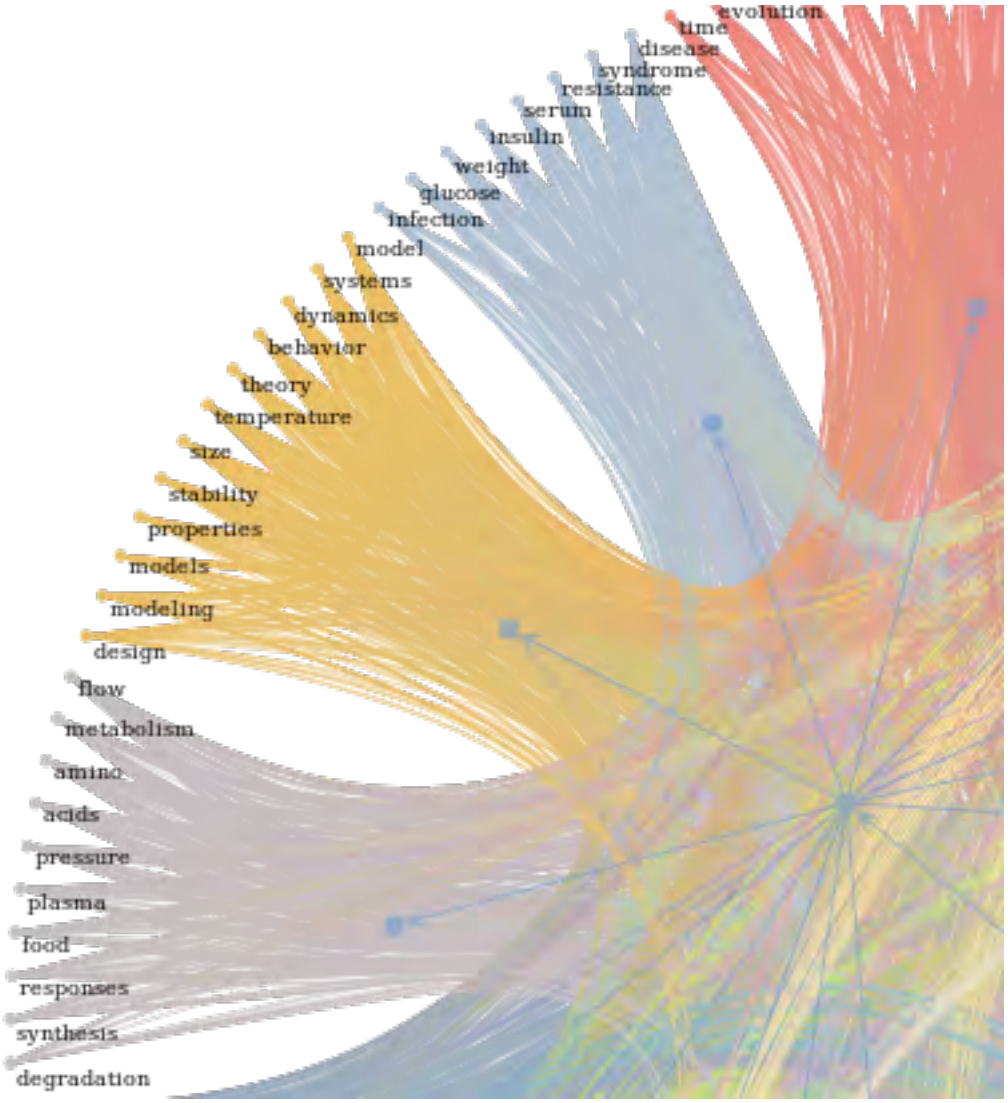


Рис. 4.10: Кластеризованный граф ключевых слов. Вложенная модель. Масштаб: три кластера, определяющие различные аспекты здоровья. На предыдущем слое иерархической кластеризации они были объединены в один кластер.

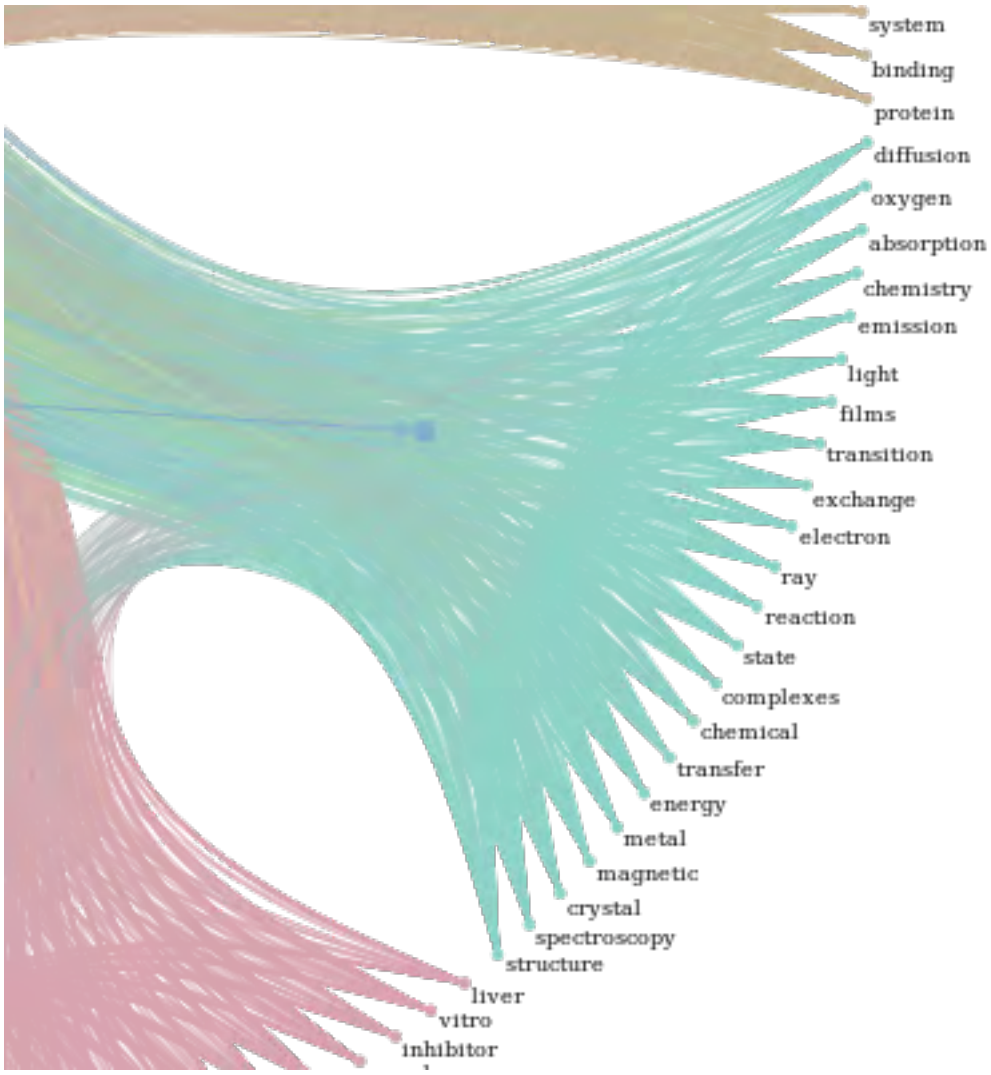


Рис. 4.11: Кластеризованный граф ключевых слов. Вложенная модель. Масштаб: выделена группа слов их химии и молекулярной физики.

Глава 5

Заключение

В ходе решения задачи исследовательской работы на данных по научным статьям были решены следующие задачи:

- Построение реляционной модели для коллекции научных статей. Написан сценарий приведения данных к третьей нормальной форме баз данных, а также сценарий загрузки нормализованных данных в базу;
- Разработка информационной панели для экспериментальных данных. Создано приложение в аналитической системе;
- Решение аналитических задачи по данным научных статей. Проведена визуализация результатов;
- Кластеризация и визуализация графов соавторства стран и ключевых слов в коллекции научных статей. Рассмотрен эффективный алгоритм стохастического блокового моделирования. В соответствии с научным взаимодействием стран раскрашена карта мира.

Литература

[1] Кузнецов С. Д. Основы современных баз данных //М: Центр Информационных Технологий. – 1998.

[2] Pover K. Learning QlikView Data Visualization. – Packt Publishing Ltd, 2013.

[3] Drake J. D., Worsley J. C. Practical PostgreSQL. – "O'Reilly Media, Inc. 2002.

[4] Paul W. Holland, Kathryn Blackmond Laskey, Samuel Leinhardt, "Stochastic blockmodels: First steps", Social Networks Volume 5, Issue 2, Pages 109-137 (1983)

[5] Brian Karrer, M. E. J. Newman "Stochastic blockmodels and community structure in networks", Phys. Rev. E 83, 016107 (2011)

[6] Tiago P. Peixoto, "Nonparametric Bayesian inference of the microcanonical stochastic block model", Phys. Rev. E 95 012317 (2017)

[7] Tiago P. Peixoto, "Parsimonious module inference in large networks", Phys. Rev. Lett. 110, 148701 (2013)

[8] Tiago P. Peixoto, "Hierarchical block structures and high-resolution model selection in large networks", Phys. Rev. X 4, 011047 (2014)

[9] Tiago P. Peixoto, "Model selection and hypothesis testing for large-scale network models with overlapping groups", Phys. Rev. X 5, 011033 (2016)

[10] Tiago P. Peixoto, "Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models", Phys. Rev. E 89, 012804 (2014)