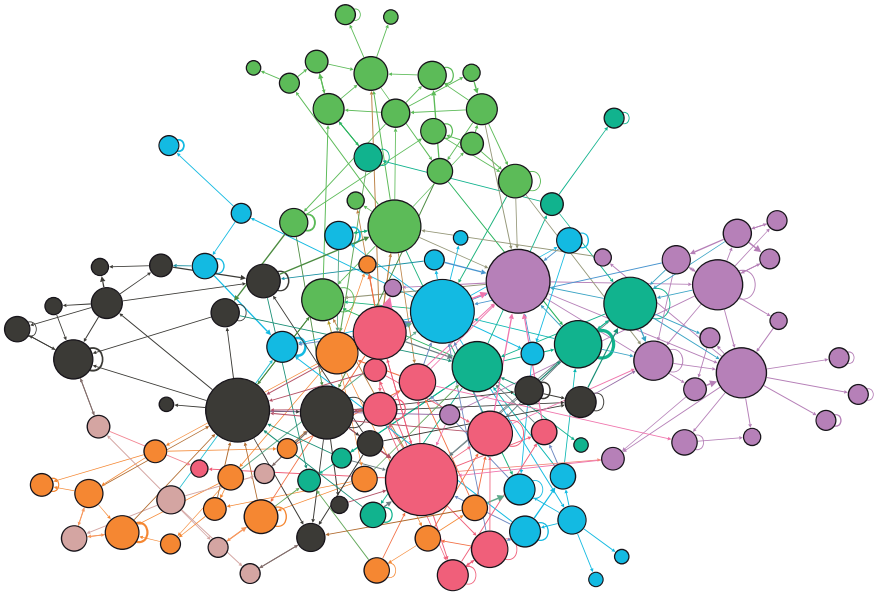
Содержание

Предисловие

Сети присутствуют во всех аспектах нашей жизни: круг друзей, коммуникационные и транспортные сети, а также Веб как Всемирная паутина – все это примеры, которые мы воспринимаем внешне, тогда как нейроны в нашем мозге и белки в нашем теле образуют сети, которые определяют наш интеллект и выживание. Когда люди общаются в Facebook или Twitter, покупают что-то в Amazon, ищут в Google или покупают авиабилет, чтобы навестить семью, они используют сети, не осознавая того. Сегодня базовое понимание сетевых процессов требуется в различных сферах деятельности – от технологий до маркетинга, от менеджмента до дизайна, от биологии до искусства и гуманитарных наук. В этом учебнике проводится разведывательный анализ учения о сетях и то, как сети помогают нам понимать сложные шаблоны взаимоотношений, которые формируют наши жизни.



Эта книга тоже является сетью! На приведенном выше рисунке показаны взаимоотношения между главами, разделами и подразделами. Связи представляют и иерархическую структуру книги (как показано в Оглавлении), и перекрестные ссылки между главами, разделами,

рисунками, таблицами, уравнениями и вставками. Цвета узлов представляют главы, а размер узла пропорционален числу соседей.

Зачем нужен «Вводный курс» по науке о сетях?

Это не первая книга о науке о сетях – на самом деле есть несколько отличных книг на выбор, и мы перечислим некоторые из них в главе 1. Мы преподаем эти темы уже в течение нескольких лет в Университете Индианы для широкой аудитории студентов старших курсов по информатике, теории вычислительных машин, науке о данных, теории информации, бизнесу, естественным и социальным наукам. Этот опыт научил нас тому, что студенты стремятся «пачкать свои руки» выполнением черновой работы и писать исходный код, чтобы понимать и использовать сети в интересующих их областях применения, даже если они только учатся программировать и не имеют математического и компьютерного образования за пределами средней школы и курсов начального уровня колледжей. Поэтому мы разработали широкий круг учебно-практических занятий и задач, как теоретических, так и вычислительных, предоставив студентам обилие практических занятий по науке о сетях. Используя такой подход, книга знакомит с сетями широкую аудиторию студентов, не имеющих никаких технических предпосылок, кроме какого-то вводного программирования и готовности учиться на деле. Это делает наш учебник пригодным для «вводного курса» науки о сетях.

Синопсис

После проведения обзора сетей, существующих во многих областях человеческих знаний и деятельности, мы поговорим о социальных сетях, которые знакомы студентам больше всего. Это позволяет ввести такие понятия, как маломировое свойство (короткие пути) и кластеризация (треугольники и транзитивность). Указанные темы объясняются с использованием увлекательных учебных занятий, таких как игра «*Шесть степеней Кевина Бейкона*». Затем мы погрузимся в роль хабов, используя Парадокс дружбы, и обсудим тему устойчивости сетей. Далее мы вводим соответственно направленные и взвешенные сети. Всемирная паутина, «Википедия», цитирование, трафик и Twitter используются для иллюстрации роли направления и веса. Последние три главы охватывают более сложные темы, а именно модели возникновения сетей, методы обнаружения сообществ и динамические процессы, происходящие в сетях.

В каждой главе рассматриваются базовые концепции, необходимые для понимания фундаментального аспекта сетей; избегаются

сложные темы и формализм. Когда это полезно, мы включаем немного математики во вставки, обрамленные рамкой. В них находится чуть-чуть более технический материал, и его можно пропустить без потери базового понимания темы. Но студенты, которые будут следить за этими дополнительными примечаниями, смогут получить более глубокое понимание материала. Каждая глава включает в себя учебно-практические занятия по программированию и упражнения, позволяющие читателям применять и проверять свои знания с помощью практических занятий по строительству и анализу сетей. Указанные учебно-практические занятия основаны на примерах реально существующих сетей, которые используются для иллюстрации концепций на протяжении всей книги. И учебно-практический исходный код, и сетевые данные доступны в репозитории книги на GitHub[1].

Целевая аудитория

С ростом популярности и коммерческого успеха онлайновых социальных сетей многие студенты заинтересованы в том, чтобы узнать немного о том, что находится «под капотом» таких сетей. Данный учебник предназначен для всех этих студентов в основном на уровне бакалавриата, хотя книга, возможно, будет полезна и для вводных курсов аспирантуры в нетехнических областях. Студенты, обучающиеся по таким программам, как наука о данных, информатика, бизнес, теория вычислительных машин, машиностроение, теория информации, биология, физика, статистика и социальные науки, получат пользу от курсов, основанных на этом учебнике. Их интерес будет достаточно велик, чтобы изучить науку о сетях глубже, и, возможно, они выберут карьеру, которая поможет найти им работу в Google, Facebook, Twitter или организовать свой собственный сетевой стартап.

Педагогика

Настоящий курс не требует никакого технического образования в области математики или программирования, что делает эту книгу пригодной для вводных курсов любого уровня, включая курсы сетевой грамотности и грамотности в программировании. Подобного рода курсы могут пропускать математические вставки. Отрабатывая учебно-практические занятия по программированию в коллаборативной вычислительной лаборатории и назначая упражнения по программированию, преподаватели предоставят студентам возможность приобрести технические навыки, достаточные для выполнения задач

[1] См. github.com/CambridgeUniversityPress/FirstCourseNetworkScience.

анализа данных, связанных с сетями. Таков наш подход в Университете Индианы, где мы преподаем материал данной книги в течение двух курсов: первый вводный курс, предназначенный для студентов-второкурсников / младших курсов, которые прошли или проходят курсы конкурентного программирования на Python; и второй курс, предназначенный для студентов младших/старших курсов. Первый курс примерно охватывает материал глав с 0 по 4. Второй курс сосредоточен на главах 5–7 после расширенного обзора и нескольких более продвинутых учебно-практических занятий по предыдущему материалу.

Обширные учебно-практические занятия по программированию и упражнения позволят преподавателям легко руководить учебным процессом и проводить практические мероприятия, а также позволят студентам укрепить и проверить свое понимание сетевых концепций. Мероприятия включают учебно-практические занятия по *NetworkX*, широко распространенной библиотеке для сетевой аналитики; и по всем затронутым в книге темам, от базовых упражнений до передовых методов. Например, на одном учебно-практическом занятии студенты знакомятся с шагами извлечения данных социальных сетей из Всемирной паутины. Используя интерфейс прикладного программирования (API) Twitter, студенты смогут анализировать популярные темы, выявлять влиятельных пользователей и реконструировать сети диффузии информации, показывающие процесс онлайнового распространения хештегов. Студенты, которые проходят учебно-практические занятия и выполняют упражнения по программированию, наберутся опыта в строительстве, импортировании/экспортировании, анализировании, манипулировании и визуализировании сетей любого типа.

Учебно-практические занятия основаны на языке Python, т. е. самом популярном языке для написания скриптов/программ. Учебное руководство, в котором рассматриваются главные концепции программирования на Python, включен в приложение А книги. Все учебно-практические занятия доступны онлайн в виде блокнотов Jupyter/IPython. Со временем библиотека NetworkX (и даже язык Python), возможно, эволюционирует, и, возможно, потребуется обновить часть исходного кода книги. Мы будем отмечать такие обновления в репозитории книги на GitHub.

Разумеется, для программирования сетей существуют и другие библиотеки, например *igraph*, *SNAP* и *graph-tool*. Наш выбор библиотеки NetworkX основан на том факте, что она написана на чистом Python, что облегчает отладку для студентов, знакомых с Python. Многие альтернативы имеют интерфейсы Python, но написаны на языке C, что делает их эффективнее, но и сложнее в отладке.

Наконец, в некоторых главах используются интерактивные модели для демонстрации сетевых явлений, таких как гигантские компоненты, малые миры, алгоритм PageRank, предпочтительное прикрепле-

ние и эпидемии. Эти модели работают в NetLogo, популярной симуляционной платформе. Учебно-практический материал по NetLogo и несколько наиболее актуальных моделей представлены в приложении В книги.

Об обложке

Сеть на обложке, сгенерированная Онуром Варолом (Onur Varol) (Феррара и соавт., 2016), изображает диффузию хештега #SB277 в Twitter. Этот хештег относится к калифорнийскому закону 2015 года об обязательствах в отношении вакцинации и освобождении от нее, и указанная сеть изображает обсуждение, которое проходило онлайн среди сторонников и противников указанного законопроекта. Узлы изображают пользователей Twitter, а связи показывают информацию, распространяемую среди пользователей через ретвиты. Размер узла отражает влияние учетной записи (сколько раз пользователь ретвитнул), а цвета узлов иллюстрируют баллы ботов: красные узлы, скорее всего, являются учетными записями ботов, синие узлы, скорее всего, являются людьми.

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге, – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв на нашем сайте www.dmkpress.com, зайдя на страницу книги и оставив комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com; при этом укажите название книги в теме письма.

Если вы являетесь экспертом в какой-либо области и заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Скачивание исходного кода примеров

Скачать файлы с дополнительной информацией для книг издательства «ДМК Пресс» можно на сайте www.dmkpress.com на странице с описанием соответствующей книги.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы обеспечить высокое качество наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг, мы будем очень благодарны, если вы сообщите о ней главному редактору по адресу dmkpress@gmail.com. Сделав это, вы избавите других читателей от недопонимания и поможете нам улучшить последующие издания этой книги.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательства «ДМК Пресс» и Manning Publications очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконной публикацией какой-либо из наших книг, пожалуйста, пришлите нам ссылку на интернет-ресурс, чтобы мы могли применить санкции.

Ссылку на подозрительные материалы можно прислать по адресу электронной почты dmkpress@gmail.com.

Мы высоко ценим любую помощь по защите наших авторов, благодаря которой мы можем предоставлять вам качественные материалы.

Введение

Сеть: взаимосвязанная или взаимодействующая цепочка, группа или система.

Вообразите мир, в котором у людей нет друзей. Где дороги никогда не пересекаются. Где компьютеры не связаны между собой. Этот мир без сетей был бы очень грустным и скучным местом, где ничего не происходит, – и даже если бы что-то случилось, никто бы об этом не узнал. Такой мир невообразим, потому что наша жизнь полностью определяется сетями: взаимоотношениями, взаимодействиями, каналами связи и Всемирной паутиной. Биологические сети, управляющие взаимодействиями между генами в наших клетках, определяют наше развитие, нейронные сети в мозге наделяют нас возможностью думать, информационные сети направляют наши знания и культуру, транспортные сети позволяют двигаться, а социальные сети подпирают нашу жизнь.

Сети – это общий, но мощный способ представления и изучения простых и сложных взаимодействий. В этой книге проводится разведывательный анализ учения о сетях и того, как они помогают нам понимать закономерности соединений и взаимоотношений, которые формируют нашу жизнь. По своей сути сеть – это простейшее описание множества взаимосвязанных сущностей, которые мы называем *узлами*, и их соединений, которые мы называем *связями*. Сетевое представление является столь общим и мощным, потому что оно устраняет многие детали конкретной системы и фокусируется на взаимодействиях между ее элементами. Отсюда сети используются для изучения самых разнообразных систем. Узлы могут представлять все виды сущностей: людей, города, компьютеры, веб-сайты, концепции, клетки, гены, виды животных и т. д. Связи представляют взаимоотношения или взаимодействия между этими сущностями: дружеские связи между людьми, рейсы между аэропортами, пакеты, которыми компьютеры обмениваются в интернете, связи между страницами Всемирной паутины, синапсы между нейронами и т. д.

Прежде чем мы представим базовые понятия, определения и номенклатуру сетей, давайте рассмотрим несколько примеров социальных, инфраструктурных, информационных и биологических сетей. Данные для всех представленных здесь примеров доступны в репозитории книги на GitHub[1]. Сети, на которых мы сосредоточимся

[1] См. github.com/CambridgeUniversityPress/FirstCourseNetworkScience.

в этой книге, как правило, являются крупными, хотя многому можно научиться, изучая и меньшие системы, такие как социальные сети, созданные на основе опросов или собеседований. В этих случаях имеет смысл детально проинспектировать отдельные узлы и соединения, тогда как анализ крупных сетей, как правило, фокусируется на макроскопических свойствах, классах узлов и связей, типичных проявлений поведения и аномалиях.

0.1. Социальные сети

Социальная сеть – это группа людей, связанных каким-либо типом взаимоотношения. Дружба, сотрудничество, романтика или простое знакомство – все это примеры социальных взаимоотношений, которые соединяют пары людей. Когда мы говорим о социальной сети, мы обычно думаем об определенном типе взаимоотношения. Человек представляется узлом в социальной сети, а взаимоотношение представляется связью между двумя людьми. Таким образом, сеть является представлением взаимоотношения. Это позволяет нам говорить о взаимоотношениях, описывать их и анализировать на уровне, выходящем за рамки пары людей.

Существует много разных типов социальных сетей, и их важно изучать. Медицинские работники анализируют сети сексуальных отношений, чтобы отыскивать способы борьбы с распространением заболеваний, передающихся половым путем. Экономисты изучают сети направления на работу для решения проблемы неравенства и сегрегации на рынках труда. А ученые инспектируют сети соавторства в научных публикациях, чтобы выявлять влиятельных мыслителей и идеи.

В наши дни мы используем веб-сайты онлайновых социальных сетей, чтобы отслеживать социальные связи. Такие платформы, как Facebook и Twitter, позволяют нам поддерживать связь со многими людьми – партнерами, друзьями, коллегами и знакомыми, иногда сотнями, – и комфортно с ними общаться независимо от расстояния. На рис. 0.1 показана сеть знакомств, часть социального графа Facebook. В этой сети узлами являются люди с учетной записью Facebook в университетах США, и соединения могут представлять различные типы взаимоотношения, от настоящей дружбы до простого знакомства. Просто взглянув на визуализацию сети, вы узнаете кое-что о лежащей в ее основе социальной структуре. У некоторых людей связей больше; мы представляем это, делая соответствующие узлы больше и темнее. Это могут быть популярные студенты, преподаватели или администраторы. Мы также замечаем, что сеть примерно поделена на две части. Данные анонимны, поэтому мы не можем сказать наверняка, но возможной интерпретацией будет то, что крупная подсеть

включает в себя в основном студентов старших курсов, а меньшая – в основном аспирантов. Между узлами в двух группах есть соединения, но их не так много, как между узлами внутри каждой группы. Другими словами, студенты старших курсов с большей вероятностью будут дружить с другими студентами, чем с аспирантами. Позже для всех этих наблюдений, которые типичны для большинства социальных сетей, мы введем формальные названия.

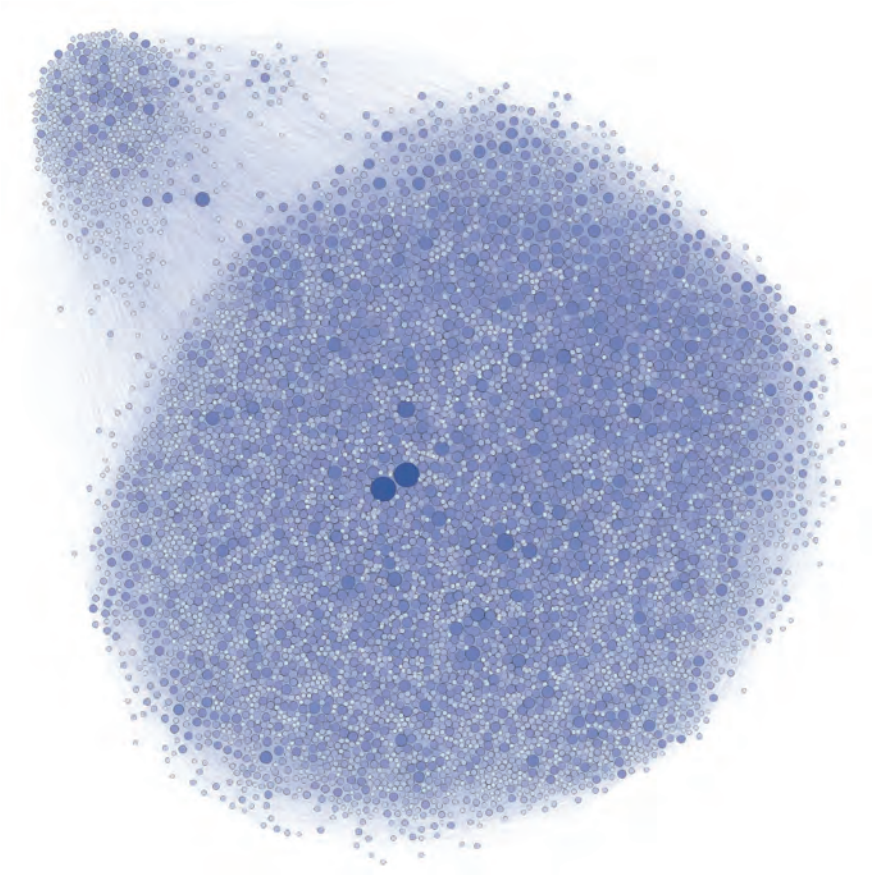


Рис. 0.1 Визуализация сети пользователей Facebook в Северо-Западном университете. Узлы обозначают людей, а связи обозначают соединения между друзьями в Facebook

Доступность данных из онлайновых социальных сетей очень увлекает ученых. Мы можем изучать человеческие взаимодействия в масштабе и в разрешающей способности, которые никогда не были возможны в прошлом: кто с кем дружит, кто на что обращает внимание, кому что нравится, что рекомендуется и как эта информация распространяется по сети. Эти данные предоставляют нам беспрецедентную возможность обнаруживать, отслеживать, добывать и моделировать

то, что делают люди. Подобно тому, как телескоп позволил нам впервые увидеть далекие планеты и звезды, а микроскоп – заглянуть в живые ткани и микроорганизмы, социальные сети позволяют изучать социальные системы и человеческую деятельность. Однако какими бы захватывающими ни были эти возможности для исследователей, они не обходятся без риска злоупотреблений. Онлайновые взаимодействия раскрывают нашу приватную информацию. Мы все слышали истории о том, как работодатели находили неловкие фотографии потенциальных сотрудников или о скандалах, имеющих отношение к хакерам и политическим организациям, собирающим данные о миллионах пользователей. Опасности бывают едва уловимыми. Обладание небольшим объемом информации о большом числе людей может раскрывать гораздо больше, чем предполагалось. Используя данные из Facebook, два студента Массачусетского технологического института обнаружили, что, просто взглянув на пол и сексуальность онлайновых друзей человека, они могут предсказывать, является этот человек геем или нет. Онлайновые социальные сети также облегчают выдачу себя за другого человека и затрудняют ее обнаружение. Выуживание информации из социальных сетей (социальный фишинг) – это метод выставления себя за друга жертвы (логически выводимого из данных онлайновой социальной сети), чтобы побуждать жертву раскрывать конфиденциальную информацию. Два студента Университета Индианы продемонстрировали, что таким образом им удалось получить секретные пароли 72 % жертв.

Данные о социальной сети можно извлекать из многочисленных источников. Если мы хотим картировать шаблоны мобильности людей с целью улучшения городских транспортных сетей, то мы можем собирать данные о звонках с мобильных телефонов. Если хотим картировать соавторство среди ученых, то можем извлекать имена из базы данных научных публикаций; два соавтора одной и той же статьи будут связаны друг с другом. (Это не тривиальное упражнение, потому что у нескольких ученых могут быть общие имена.) Если мы хотим картировать сотрудничество между кинозвездами, можем извлекать данные о титрах кинофильмов из интернет-базы данных кинофильмов (Internet Movie Database, IMDB.com). На рис. 0.2 показаны две такие сети. В одном случае на самом деле существует два вида узлов: кинофильмы и актеры/актрисы. Мы проводим связь между актрисой и кинофильмом, в котором она снялась. В другом случае мы фокусируемся на связях между актерами/актрисами, которые снимались в фильмах вместе. Хотя изображенные сети улавливают лишь крошечные части базы данных кинофильмов, мы снова замечаем некоторые четкие закономерности. Более крупные узлы имеют больше соединений, представляющих звезд, которые снимались во многих кинофильмах. Мы также видим, что сети структурированы в несколько плотных групп, связанных с периодами, языками или жанрами кинофильмов: голливудские (синие), европейские (голубые), мексиканские (фиолетовые), китайские (желтые), филиппинские (оранжевые),

турецкие и восточноевропейские (зеленые), индийские (красные), греческие (белые) кинозвезды и кинозвезды фильмов для взрослых (розовые) на рис. 0.2(b). В главе 6 вы узнаете, как обнаружить эти группы и выяснять, чему они посвящены.

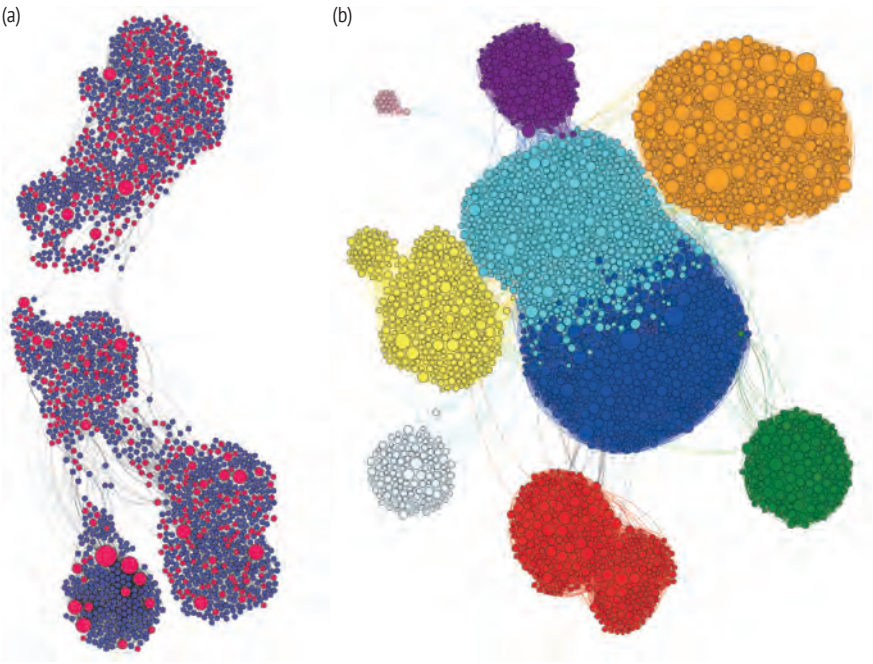


Рис. 0.2 (a) Сеть кинозвезд, основанная на небольшой выборке кинофильмов, актеров и актрис из интернет-базы данных кинофильмов. Узлы представляют кинофильмы (синие) или актеров/актрис (красные). Связь соединяет актера или актрису с фильмом, в котором они снимались. (b) Сеть киносозвездий, основанная на небольшой выборке актеров и актрис из интернет-базы данных кинофильмов, снимавшихся вместе с другой кинозвездой. Связь соединяет двух людей, которые снялись вместе по меньшей мере в одном фильме. Цвета представляют жанры фильмов или языки/страны

0.2. Коммуникационные сети

В сетях Facebook и кинофильмов связи взаимны: вы не можете подружиться с кем-либо на Facebook, если он не согласен, и вы не можете быть снятым в фильме, не будучи упомянутым в титрах. Однако не все социальные сети имеют взаимные связи. Например, Twitter представляет собой популярную социальную сеть со связями, которые не обязательно являются взаимными: Алиса может следить за Бобом без того, чтобы Боб обязательно следил за Алисой. Как следствие, запечатленные в сети Twitter отношения не являются дружбой; вы подписываетесь на кого-то, чтобы увидеть, что он публикует. Когда вы ретвитите сообщение, его видят ваши подписчики. Это хороший

способ широко обмениваться информацией, поэтому Twitter является социальной сетью, в основном направленной на распространение информации, т. е. коммуникационной сетью. Ретвитная сеть на рис. 0.3 иллюстрирует распространение политических сообщений во время выборов в США. Более крупные узлы являются узлами с большим числом исходящих связей, потому что число ретвитов пользователей другими пользователями является способом измерить их влияние. Вы, вероятно, сразу заметили более поразительную закономерность: консервативные пользователи (красные узлы) в основном ретвитят сообщения от других консерваторов, в то время как прогрессивные пользователи (синие узлы) аналогичным образом делятся прогрессивным контентом. На самом деле такие предпочтительные регулярности социальных связей позволяют нам с высокой точностью угадывать политические склонности человека. Это свойство, именуемое *гомофилией*, будет обсуждаться в главе 2; алгоритм определения политических предпочтений по структуре сети будет представлен в главе 6.

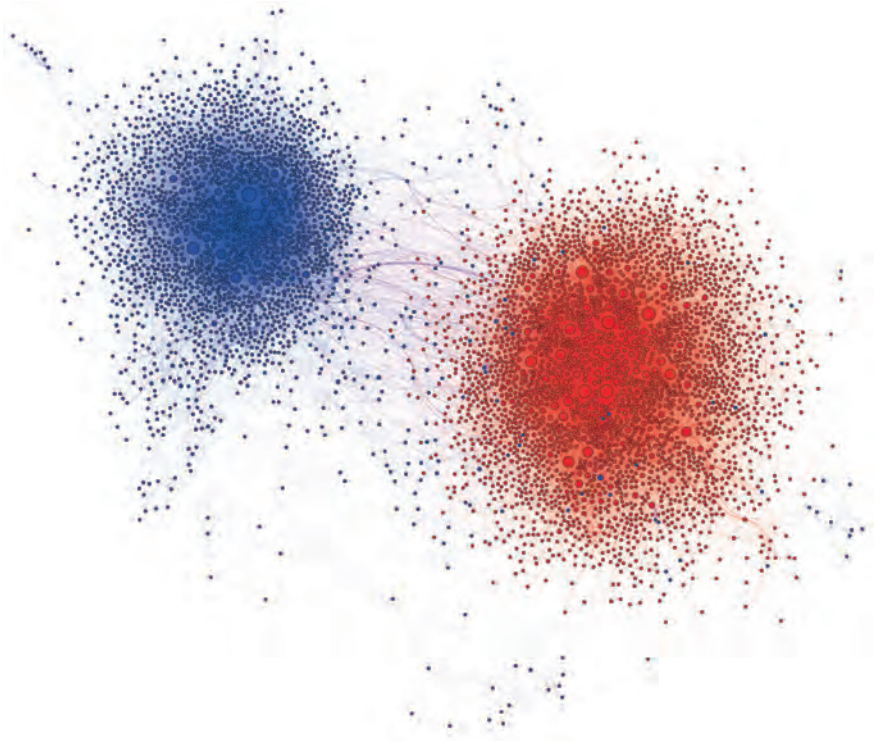


Рис. 0.3 Ретвитная сеть в Twitter среди людей, делящихся постами о политике США. Связи представляют собой ретвиты сообщений, в которых использовались хештеги, такие как `#tcot` и `#p2`, связанные соответственно с консервативными (красными) и прогрессивными (синими) сообщениями во время промежуточных выборов в США 2010 года. Когда Боб ретвитит Алису, мы рисуем направленную связь от Алисы к Бобу, чтобы обозначить, что сообщение перешло от нее к нему. Направление связей не показано

Такие сети, как Twitter, позволяют нам отслеживать диффузию хештегов и новостей, наблюдая за тем, как идеи и культурные концепции распространяются от человека к человеку. Но социальные сети также используются для распространения дезинформации, которая неосознанно передается доверчивыми пользователями. Используя поддельные новостные веб-сайты и автоматизированные или полуавтоматические учетные записи, именуемые «социальными ботами», вредоносная организация может дешево и эффективно генерировать и усиливать кампанию по дезинформации в политических целях либо для монетизации трафика с помощью рекламы. В последние годы мы наблюдаем резкий рост подобных видов манипуляций социальными сетями в глобальном масштабе. Если кто-то может контролировать информацию, которую люди видят онлайн, то он может манипулировать их мнением. Во многих странах это явление представляет угрозу демократии, потому что без хорошо информированных избирателей невозможно проводить свободные выборы. Академические исследователи и промышленные инженеры усердно работают над разработкой контрмер. Понимание структуры и динамики сетей, обеспечивающих распространение информации, является важнейшим компонентом этих усилий.

Социальные связи в Twitter создаются до того, как пользователь создает сообщение, которое обычно транслируется всем подписчикам пользователя. В электронной почте, как и в социальных сетях, узлы являются людьми. Однако каждое сообщение предназначено для одного или нескольких конкретных получателей. Связи основаны на обмениваемых сообщениях. Электронная почта не зависит от конкретной платформы; протокол открыт и распространяется, вследствие чего ни одна организация не контролирует весь трафик. Как следствие, электронная почта по-прежнему остается одной из наиболее широко используемых коммуникационных сетей. На рис. 0.4 показан пример сети электронной почты. Опять же, связи направляются от отправителя к получателю электронного письма обозначенными стрелками. Размер и цвет узла представляют два разных признака: число соответственно входящих и исходящих связей. Более крупный узел получает электронные письма от большего числа отправителей, а более темный узел отправляет электронные письма большему числу получателей. Тот факт, что более крупные узлы, как правило, темнее и наоборот, говорит нам о том, что между отправкой и получением электронных писем существует корреляция.

Рис. 0.4 Сеть, опирающаяся на базу данных электронных писем, сгенерированных сотрудниками энергетической компании Enron. Эти данные были получены Федеральной комиссией по регулированию энергетики США в ходе расследования, проведенного после краха компании в 2001 году. По завершении расследования электронные письма были признаны как находящиеся в публичном пространстве и сделаны общедоступными для исторических исследований и академических целей. Показана только небольшая часть центрального ядра сети. Направление связей показано стрелками

0.3. Всемирная паутина и «Википедия»

Всемирная паутина (Веб) – это крупнейшая информационная сеть. Хотя сейчас она используется для предоставления всех видов услуг, изначально это была просто сеть документов (страниц), соединенных «гиперсвязями», или кликабельными связями. В начале 1990-х годов Тим Бернерс-Ли захотел упростить доступ ученых к информации об экспериментах по физике высоких энергий в Европейской организации ядерных исследований (CERN) недалеко от Женевы. Он выдвинул три ключевые идеи: (1) систему именования страниц, Единый локатор ресурсов (Uniform Resource Locator, URL); (2) простой язык для написания документов, именуемый языком разметки гипертекста (HyperText Markup Language, HTML), включая гиперсвязи из одной страницы на другую; и (3) простой протокол, именуемый протоколом передачи гипертекста (HyperText Transfer Protocol, HTTP), для программ-клиентов (браузеров), чтобы общаться с серверами. Благодаря этим трем компонентам родилась Всемирная паутина. Бернерс-Ли даже имплементировал первый веб-сервер и программно-информационное обеспечение для браузера, чтобы скачивать страницы и мультимедиа с серверов, нажимая на связи. На самом деле мы можем видеть здесь участие двух сетей: статический «граф связей», состоящий из моментального снимка веб-страниц и связей в данный момент времени,

и динамическую сеть трафика, возникающую в результате передвижения людей по сети. Перефразируя классическую философскую загадку, если между двумя страницами есть связь, но никто на нее не нажимает, действительно ли она является частью паутины? Ответ, конечно, зависит от того, о какой из двух сетей мы думаем, когда произносим слово «паутина». В последующих главах мы потратим больше времени на разведывание обеих этих информационных сетей.

Всемирная паутина слишком велика, чтобы визуализировать даже малую ее часть осмысленным образом. Давайте сосредоточимся на «Википедии», которая представляет собой сеть страниц (статей) на одном веб-сайте. «Википедия» – это коллаборативная энциклопедия, редактируемая тысячами добровольцев по всему миру, и это одно из самых популярных направлений во Всемирной паутине. Существуют версии «Википедии» на многих языках, поэтому давайте сосредоточимся на английской. Тем не менее английская «Википедия» представляет собой огромную сеть с миллионами статей (и она растет!). Поэтому давайте сосредоточимся лишь на небольшом подмножестве статей по математике, показанном на рис. 0.5. Здесь размер узла представляет метрику *PageRank*, меру значимости, отражающую степень важности статьи на основе других статей, которые имеют с ней связь, – тему нашего обсуждения в главе 4. Например, крупный белый узел посередине – это общая статья по *математике*. Еще одним при-

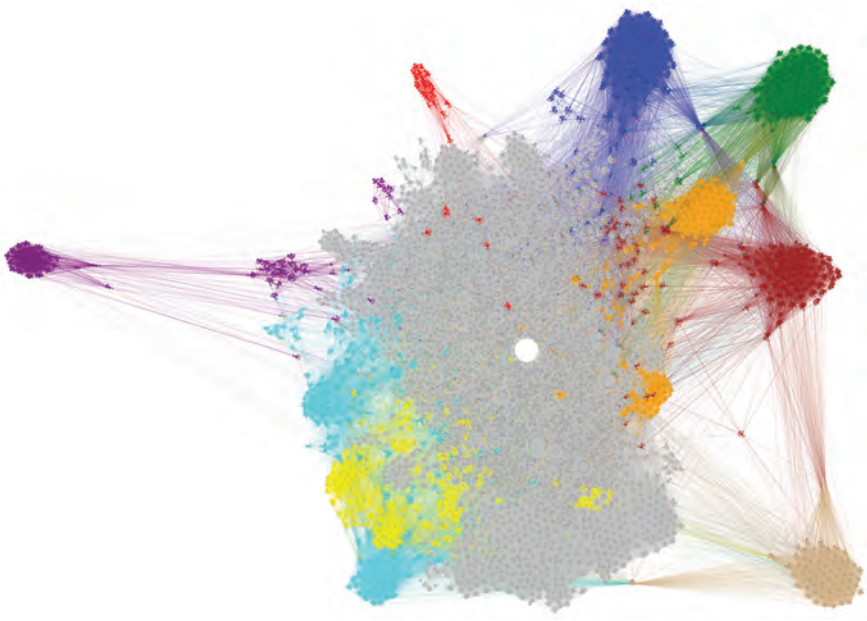


Рис. 0.5 Часть информационной сети «Википедии». Узлы – это статьи о математике. Мы рассматриваем связи только между статьями «Википедии» и игнорируем связи с внешними страницами. Размер узла пропорционален важности статьи, а цвета выделяют сообщества, обсуждаемые в тексте

знаком этой сети является наличие крупного «ядра» (серого цвета) и нескольких малых групп. Эти группы представляют собой тесно связанные группы статей по конкретным темам или разделам математики. Например, статьи об исторических греческих (синих), арабских (зеленых) и индийских (коричневых) математиках; о современных индийских математиках (коричневый); о математике и искусстве (оранжевый), статистике (голубой), теории игр (желтый), математическом программно-информационном обеспечении (фиолетовый) и педагогической теории (красный). Мы также наблюдаем несколько «мостовых» узлов, которые соединяют несколько кластеров. Эти признаки можно найти во многих реально существующих сетях.

0.4. Интернет

Мы часто думаем об интернете как о сети компьютеров и других присоединенных устройств, но в реальности это *сеть сетей*. На самом деле это слово происходит от английского слова internetworking, т. е. *межсетевое взаимодействие*, или соединения разных компьютерных сетей через специальные узлы, именуемые *маршрутизаторами*. И по этой причине мы можем наблюдать интернет на многих уровнях: на самом низком уровне у нас аппаратные устройства, которые соединяют отдельные компьютеры в одну локальную или широкомасштабную сеть. Эти сети соединяются маршрутизаторами, поэтому мы можем уменьшать масштаб и думать о сети маршрутизаторов. Если мы еще больше уменьшим масштаб, то обнаружим группы сетей, управляемых провайдером интернет-служб (Internet Service Provider, ISP). Эта организация определяет свою внутреннюю сетевую топологию (способ соединения маршрутизаторов) самостоятельно и поэтому также называется «автономной системой» (AS). Специальные «пограничные» маршрутизаторы соединяют одну автономную систему с другой, образуя то, что мы называем сетью автономных систем.

На рис. 0.6 показана небольшая часть сети интернет-маршрутизаторов. Хотя интернет развивался без централизованного контроля или координации, провайдеры интернет-служб соблюдают локальные правила соединения своих маршрутизаторов. Они стараются обеспечивать наилучшее обслуживание по самой низкой цене. В результате возникают определенные регулярности. Например, та часть интернета, которая обеспечивает наибольший трафик, часто называется «магистралью». Крупные телекоммуникационные компании, управляющие интернет-магистралью, заинтересованы в предотвращении сбоев, поэтому они конструируют свои сети с большой избыточностью. Осюда мы наблюдаем плотное «ядро», в котором крупные маршрутизаторы соединены друг с другом. По мере того как мы продвигаемся к «периферии» интернета – нашим домашним маршрути-

заторам, – сеть становится соединенной все более разреженно. Подобного рода иерархическая *структура ядро–периферия* распространена в многочисленных разных типах сетей и будет обсуждаться в главе 2. В сети маршрутизаторов, изображенной на рис. 0.6, зеленый кластер слева хорошо сепарирован от остальной сети. Вероятно, это обусловлено систематическим смещением в методологии зондирования, используемой для картирования этих сетей: большинство измерений было проведено в Соединенных Штатах, и маршрутизаторы в этом кластере расположены там. Относящейся к этому отличительной особенностью является наличие очень крупных узлов в зеленом кластере, что указывает на маршрутизаторы с большим числом соединений. На самом деле это может быть ошибкой измерения, вызванной тем же систематическим смещением. Ввиду аппаратных ограничений маршрутизатор фактически может иметь только ограниченное число соединений. Пусть это послужит напоминанием о том, что если мы используем ущербный метод сбора данных о сети, то его анализ может привести к неправильным выводам.

Рис. 0.6 Часть сети интернет-маршрутизаторов. Карта представляет собой снимок, созданный Центром прикладного анализа интернет-данных (Center for Applied Internet Data Analysis, CAIDA.org) с использованием инструментов, которые отправляют малые пакеты данных (зонды) между хостами интернета. Цвета назначаются в соответствии с алгоритмом обнаружения сообществ, который выявляет плотные кластеры, отражающие географическое распределение маршрутизаторов. В главе 6 вы узнаете, как использовать эту методологию для изучения того, что эти кластеры представляют

0.5. Транспортные сети

Еще один важный класс сетей касается различных видов транспортных перевозок. Узлами являются местоположения: города, перекрестки дорог, аэропорты, порты, железнодорожные станции или станции

метро. Однако эти сети сильно отличаются друг от друга. Например, дорожные сети развиваются локально, чтобы минимизировать расстояние, проходимое между близлежащими городами. Это приводит к появлению решетчатых структур, в которых большинство узлов имеет сопоставимое число соединений – к примеру, четырехпутные пересечения. На рис. 0.7 показана сеть авиационных перевозок, которая не имеет решетчатой структуры. Причина в том, что авиакомпании стараются минимизировать число перелетов между пунктом отправления и пунктом назначения, не добавляя дорогостоящих прямых рейсов между аэропортами с низким трафиком. Простое решение состоит в добавлении рейсов, соединяющих аэропорты с существующими хабами, действующими как транспортно-пересадочные узлы. Как следствие, сети авиарейсов имеют структуру «хаб и спица» (hub and spoke): несколько хабов имеет огромные числа связей, тогда как большинство узлов имеет очень мало соединений.

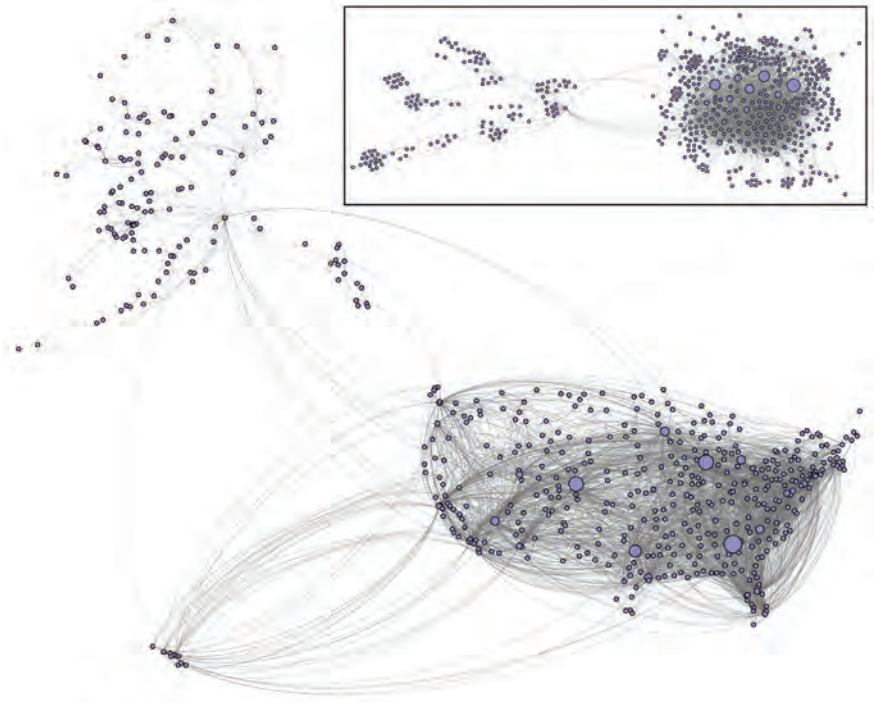


Рис. 0.7 Сеть авиационных перевозок США (данные о рейсах из OpenFlights.org). Узлы расположены в соответствии с географическими координатами соответствующих аэропортов, вследствие чего мы можем различить очертания континентальной части Соединенных Штатов, Аляски и Гавайев. Обратите внимание, что проекция карты делает Аляску больше, чем ее фактический размер, из-за ее широты. Авиационные хабы с большинством соединений (например, Атланта, Чикаго, Денвер) четко узнаваемы. Вставка отображает ту же сеть, но с другой «направляемой силой» компоновкой, обсуждаемой в разделе 1.10

При изучении определенных типов сетей, в особенности относящихся к транспортным перевозкам и коммуникациям, мы можем рассуждать о них с точки зрения их статической структуры или динамических процессов, происходящих в этих сетях. Возьмем, например, сеть авиационных перевозок. Мы могли бы рассматривать карту на рис. 0.7 как множество маршрутов, существующих между аэропортами, независимо от фактического движения по ним; или как транспортную сеть, возникающую в результате передвижения людей между аэропортами. В последнем смысле связи разнообразны, потому что они несут разный объем трафика, а также меняются со временем. Важны как структура, так и динамика сетей. Иногда мы просто улавливаем динамику, представляя трафик через направления и веса связей, как обсуждается в главе 4. В других случаях мы, возможно, захотим изучить фактические процессы, которые позволяют сети расти и меняться с течением времени, или взаимодействия, которые происходят в сети. Главы 5 и 7 посвящены этим темам, касающимся сетевой динамики.

0.6. Биологические сети

В клетках нашего организма специальные молекулы, именуемые белками, взаимодействуют различными способами. Например, когда белок сворачивается, его изменение структуры может регулировать функцию другого белка или активность фермента. Ферменты (сами по себе белки) катализируют биохимические реакции и жизненно важны для обмена веществ, который поддерживает жизнь, собирая энергию для строительства и поддержания белков, составляющих наши ткани и органы. Белки также регулируют клеточную сигнализацию и иммунные реакции. Все эти взаимодействия можно рассматривать как сети: сети белковых взаимодействий, метаболические сети, генно-регуляторные сети и т. д. Эти биологические сети существуют внутри клетки. На более высоком уровне, внутри тела, связи между нервными клетками (синапсами) приводят к возникновению нейронных сетей, которые формируют наш мозг. И на еще более высоком уровне взаимодействуют целые виды животных. Животное одного вида может рассматривать другой вид как пищу, создавая экологическую сеть или пищевую (трофическую) паутину между видами. Когда мы думаем об этой сети, экологический баланс зависит от наличия видов, которые поддерживают друг друга. Удаление узла в такой пищевой паутине – например, когда вид вымирает, – влияет на выживание других частей экосистемной сети. На рис. 0.8 показаны три типа биологических сетей: сеть белковых взаимодействий, нейронная сеть и пищевая паутина. Все они являются важнейшими элементами жизни на нашей планете.

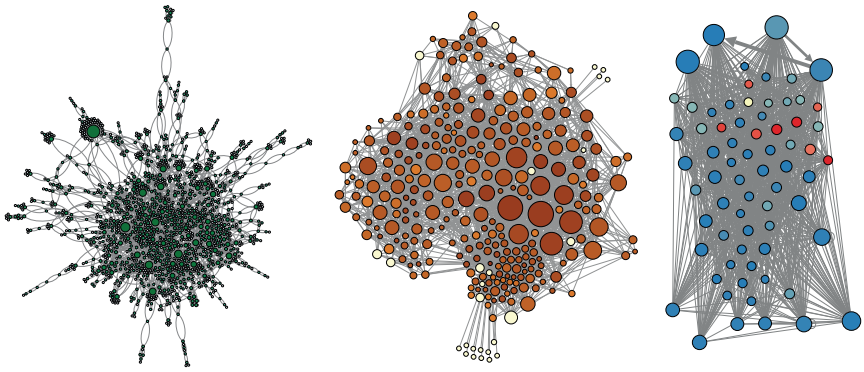


Рис. 0.8 Три биологические сети. Слева: сеть взаимодействия дрожжевых белков. Размер узла пропорционален числу взаимодействующих белков. В центре: нейронная сеть круглого червя *Caenorhabditis elegans*. Крупные и красные узлы представляют нейроны с большим числом соответственно исходящих и входящих синапсов. Справа: пищевая паутина видов в Национальном парке Эверглейдс во Флориде. Направленная связь переходит от жертвы к виду хищника. Вес (ширина) связи представляет собой поток энергии между двумя видами. Размер и цвет узлов представляют соответственно входящие и исходящие связи, вследствие чего крупные синие узлы являются видами в верхней части пищевой цепочки, в то время как малые красные узлы являются видами в нижней части

0.7. Резюме

Сети – это общий способ моделирования и изучения сложных систем со многими взаимодействующими элементами. Мы увидели несколько примеров сетей. Узлы могут представлять самые разные типы объектов, от людей до веб-страниц, от белков до видов животных, от интернет-маршрутизаторов до аэропортов. Узлы могут иметь связанные с ними признаки, помимо меток: географическое местоположение, богатство, активность, число соединений и т. д. Связи также могут представлять много разных видов отношений, от физических до виртуальных, от химических до социальных, от коммуникативных до информационных. Они могут иметь направление (например, веб-связи, так называемые веб-ссылки, и электронная почта)[1] или быть взаимными (например, брак). Все они могут быть одинаковыми или

[1] Обратите внимание, что в науке о сетях все базовые концепции трактуются через призму узлов и связей (а не ссылок) как аналогов терминов «вершины» и «ребра» из теории графов. Отсюда и термин «связь во Всемирной паутине», или «веб-связь» (Web link), и «гиперсвязь» (hyperlink) как связь особого рода, которая связывает страницу с другим ресурсом в паутине. Также следует отметить, что проводится четкое различие между Всемирной паутиной как информационной сетью и интернетом как компьютерной сетью, или сетью маршрутизаторов. – *Прим. перев.*

иметь разные признаки, такие как сходство, расстояние, трафик, объем, вес и т. д.

0.8. Дальнейшее чтение

Использование сетей для графического представления социальных взаимоотношений между индивидуумами было введено Морено и Дженнингсом (1934), которые назвали эти социальные сети социограммами.

Совсем недавно исследования показали, что онлайновые социальные сети могут выявлять сексуальную ориентацию человека (Джерниган и Мистри, 2009) и способствовать высокоэффективным фишинговым атакам (Джагатик и соавт., 2007). Коновер и соавт. (2011b) показали, что сети диффузии политической информации в Twitter очень поляризованы и сегрегированы. Как следствие, мы можем с высокой точностью предсказывать политические взгляды большинства пользователей, начиная с нескольких меток узлов и распространяя их через соседей по сети (Коновер и соавт., 2011a).

По теме видения, дизайна и истории Всемирной паутины можно прочитать в книге, написанной в соавторстве с его изобретателем (Бернерс-Ли и Фишетти, 2000).

Спринг и соавт. (2002) объясняют принцип применения зондов для измерения топологии интернета. Ахлиоптас и соавт. (2009) показывают, что эти подходы имеют систематическое смещение при взятии выборок. Ученые в области теории вычислительных машин анализируют структуру маршрутизаторов и сетей автономной системы для разработки моделей, именуемых «генераторами топологии», которые будут помогать в дизайне этих сетей (Росси и соавт., 2013). В целях более подробного ознакомления с интернет-сетями мы рекомендуем книгу Пастора-Саторраса и Веспиньяни (2007).

Данные о сети взаимодействий дрожжевых белков взяты из работы Хеонга и соавт. (2001). Данные нейронной сети C. elegans взяты из работы Уайта и соавт. (1986). В целях ознакомления с сетями человеческого мозга, или «коннектомом», мы рекомендуем работу Спортса (2012). Экологическая сеть Эверглейдс взята из работы Улановича и Деанджелиса (1998). В целях более подробного ознакомления с пищевыми паутинами мы отсылаем к работам Данна и соавт. (2002) и Мелиана и Баскомпте (2004).

Данные для нескольких примеров реально существующих сетей, показанных в этой книге, предоставлены Сетевым репозиторием (Network Repository, Росси и Ахмед, 2015). Визуализации выполняются с использованием Gephi (Бастиан и соавт., 2009). Алгоритмы компоновки обсуждаются в главе 1.

Упражнения

0.1 Рассмотрите дорожную карту на рис. 0.9. Если бы кто-то создавал сетевое представление регулярностей дорожного движения, то какой из следующих ниже вариантов был бы самым лучшим для создания связей сети? (*Подсказка:* ваш ответ на следующий далее вопрос может повлиять на ваш ответ на этот вопрос, и наоборот.)

- **a.** Движущиеся по улицам пешеходы.
- **b.** Участки дорог (например, 5-я авеню между 12-й и 13-й улицами).
- **c.** Целые дороги (например, 5-я авеню).
- **d.** Движущиеся по дорогам транспортные средства.

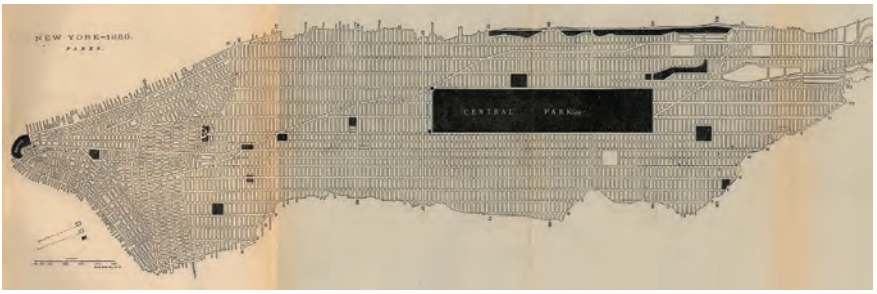


Рис. 0.9 Карта Нью-Йорка в 1880 году. Из Отчета о социальной статистике городов, составленного Джорджем Э. Уорингом-младшим (George E. Waring, Jr.), Бюро переписи населения США, 1886 год. Изображение предоставлено библиотеками Техасского университета

0.2 Рассмотрите дорожную карту на рис. 0.9. Какой из следующих ниже вариантов в сетевом представлении шаблонов дорожного движения был бы самым лучшим для создания узлов сети? (*Подсказка:* ваш ответ на предыдущий вопрос может повлиять на ваш ответ на этот вопрос, и наоборот.)

- **a.** Городские кварталы (например, квартал между 5–6-й авеню и 12–13-й улицами).
- **b.** Уличные перекрестки (например, 5-я авеню и 12-я улица).
- **c.** Движущиеся по улицам пешеходы.
- **d.** Движущиеся по дорогам транспортные средства.

0.3 Рассмотрите сеть авиационных перевозок США, показанную на рис. 0.7. Узлы в этой сети изображают аэропорты. Что могла бы представлять связь между двумя аэропортами?

0.4 Сравните сеть авиационных перевозок США на рис. 0.7 с дорожной картой Манхэттена на рис. 0.9. Сеть авиационных перевозок демонстрирует отличительный признак, которого не хватает

у дорожной сети Манхэттена. Какова эта ключевая характеристика?

a. Узлы-одиночки без связей.
b. Многочисленные маршруты между узлами.
c. Узлы с более чем одной присоединенной связью.
d. Хабовые узлы со множеством связей.

0.5 Какой тип связи лучше всего отражает взаимоотношение «друг» в социальном графе из Facebook? Направленная или ненаправленная?

0.6 Какой тип связи лучше всего отражает взаимоотношение «подписчик» в социальном графе из Facebook? Направленная или ненаправленная?

1 Сетевые элементы

Узел: точка в сети или схеме, в которой линии или пути пересекаются или разветвляются.

Рассмотрев несколько примеров реально существующих сетей в главе 0, давайте теперь познакомимся с базовыми определениями и величинами, которые позволяют нам описывать сеть.

1.1. Базовые определения

В очень общих чертах сеть, или граф, представляет собой множество элементов, которые мы называем *узлами*, а также множество соединений между парами узлов, которые мы называем *связями*. Связи обозначают наличие взаимоотношения между элементами, представленными узлами. Как мы видели ранее, связи могут соответствовать социальным, физическим, коммуникационным, географическим, концептуальным, химическим, биологическим или другим взаимодействиям. Мы говорим, что два узла являются *смежными*, *соединены*, или *связаны*, если между ними есть связь. Соединенные узлы также принято называть *соседями*.

Сети обеспечивают общий теоретический каркас, допускающий удобное концептуальное представление взаимоотношений в широком спектре систем; в главе 0 мы увидели несколько примеров таких систем. Изучение сетей имеет давние традиции в математике, информатике, социологии и исследованиях в области коммуникаций. В последнее время сети также интенсивно изучаются в физике и биологии. Разные области, которые имеют отношение к сетям, нередко вводят свою собственную номенклатуру. Например, в некоторых областях сеть называется *графом*, узел называется *вершиной*, а связь – *ребром*. (Время от времени мы будем использовать эти термины.) Строгий язык описания сетей можно найти в теории графов, области математики, которая восходит к новаторской работе Леонарда Эйлера в XVIII веке. Здесь мы не хотим давать строгого введения в теорию графов. Мы в основном заинтересованы в построении словаря и введении набора базовых понятий, которые позволят нам сделать первые шаги в мир сетей. Однако иногда полезно использовать формальную нотацию. В этих случаях мы будем включать формальную нотацию в заштрихованную область, или во вставку, обрамленную

рамкой. Например, более строгое определение сети приведено во вставке 1.1. В последующих главах мы будем вводить дополнительные понятия и определения, необходимые для анализа реально существующих систем.

Вставка 1.1 **Определение сети**

Сеть G состоит из двух частей, множества из N элементов, именуемых *узлами*, или *вершинами*, и множества из L пар узлов, именуемых *связями*, или *ребрами*. Связь (i, j) соединяет узлы i и j. Сеть может быть направленной или ненаправленной[1]. Направленная сеть также называется *орграфом* от термина «ориентированный граф». В направленных сетях связи называются *направленными связями*, и порядок узлов в связи отражает направление: связь (i, j) идет из источникового узла i в целевой узел j. В ненаправленных сетях все связи являются двунаправленными, и порядок расположения двух узлов в связи не имеет значения. Сеть может быть невзвешенной или взвешенной. Во взвешенной сети со связями ассоциированы *веса*: *взвешенная связь* (i, j, w) между узлами i и j имеет вес w. Сеть может быть как направленной, так и взвешенной, и в этом случае она имеет направленные взвешенные связи.

Каждая сеть характеризуется суммарным числом узлов N и суммарным числом связей L. Мы называем N *размером* сети, потому что оно определяет число отдельных элементов, составляющих систему. Чисел узлов и связей недостаточно для определения сети; мы должны указать способ, которым узлы соединяются связями.

Существуют разные типы связей, которые определяют разные классы сетей. В некоторых сетях, таких как Facebook (рис. 0.1), связи не имеют направления, и мы представляем их в виде отрезков. Мы называем такие сети *ненаправленными*. В других случаях, таких как «Википедия» (рис. 0.5), связи направлены, и мы представляем их в виде стрелок. Сети с направленными связями называются *направленными сетями*. Мы расскажем о направленных сетях подробнее в разделе 1.6 и главе 4.

В некоторых случаях, таких как сети авиационных перевозок (рис. 0.7), связи имеют соответствующие веса. Они называются *взвешенными сетями*. Сеть может быть как направленной, так и взвешенной. Сеть электронной почты является примером взвешенной направленной сети, в которой веса и направления связей представляют трафик связи (число сообщений) между узлами. Мы вернемся к взвешенным сетям в разделе 1.7 и главе 4. На рис. 1.1 представлены иллюстрации ненаправленных, направленных и взвешенных сетей.

[1] Понятия направленности и ориентированности связей (и сетей в целом) являются синонимичными. В переводе принят первый вариант. Понятие ориентированности чаще используется в теории графов. – *Прим. перев.*

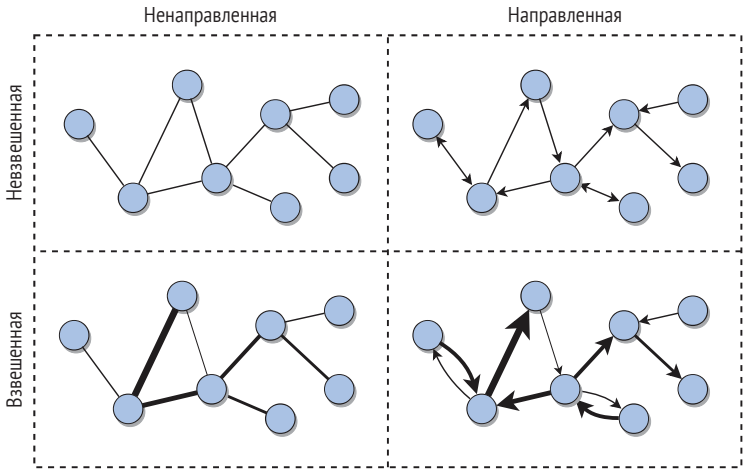


Рис. 1.1 Графические представления ненаправленных, направленных и взвешенных сетей. Круги представляют узлы. Пары смежных узлов соединяются отрезком (связью) или стрелкой (направленной связью). Стрелки указывают направление связей. Толщина связи представляет ее вес во взвешенных сетях

Существует несколько других классов сетей. В сети может быть несколько типов узлов. Например, сеть кинозвезд (рис. 0.2(а)) имеет два типа узлов, представляющих кинофильмы и людей. В этой сети связь соединяет актера или актрису с кинофильмом, но нет никаких связей между людьми или между кинофильмами. Это пример так называемой *двудольной сети*. В двудольной сети есть две группы узлов, таких, при которых связи соединяют узлы только из разных групп, а не узлы из одной и той же группы. Другие примеры двудольных сетей включают сети, которые улавливают взаимоотношения между песнями и исполнителями, между учебными занятиями и студентами, а также между товарами и покупателями. Подробнее о двудольных сетях вы узнаете в главе 4.

Сеть может иметь несколько типов связей, и в этом случае она называется *мультиплексной* сетью. Еще раз используя пример с кинозвездами, мы могли бы вообразить добавление связей между актерами и/или актрисами, которые находятся в супружеской связи друг с другом. В примере с «Википедией» (рис. 0.5) в дополнение к гиперсвязям у нас могут быть взвешенные связи, представляющие клики пользователей «Википедии», и/или ненаправленные связи между статьями, которые имеют общих редакторов. Эти и другие более сложные типы сетей обсуждаются далее в разделе 1.8.

1.2. Манипулирование сетями в исходном коде

Для управления, анализа и визуализации сетей с более чем несколькими узлами и связями нам необходимо использовать программно-

информационные инструменты или писать собственный исходный код. Существует масса инструментов сетевого анализа и визуализации, а также библиотек для работы с сетями на многих языках программирования. На протяжении всей книги мы время от времени будем упоминать пару таких инструментов. Например, визуализации в главе 0 генерируются с помощью приложения под названием *Gephi*. Однако мы считаем, что для практического понимания сетей необходимо «запачкать свои руки» выполнением черновой работы и написать немного исходного кода. Мы исходим из того, что студенты, использующие эту книгу, имеют некоторое представление о Python, популярном языке программирования как среди начинающих, так и среди опытных программистов[1]. Чтобы облегчить жизнь, мы будем использовать *NetworkX* (networkx.github.io), пакет Python для создания, управления и изучения структуры, динамики и функций сетей. NetworkX предоставляет структуры данных, алгоритмы, меры и генераторы для сетей, а также рудиментарные средства визуализации[2].

После импортирования библиотеки NetworkX мы можем легко создать ненаправленную сеть («Граф») и добавить несколько узлов и связей. Обращение к узлам осуществляется посредством целочисленных идентификаторов, а связи называются ребрами (edge):

```
import networkx as nx  # всегда сначала следует импортировать NetworkX!
G = nx.Graph()
G.add_node(1)
G.add_node(2)
G.add_edge(1,2)
```

Мы можем добавить несколько узлов или связей одновременно:

```
G.add_nodes_from([3,4,5,...])
G.add_edges_from([(3,4),(3,5),...])
```

Вот как мы получаем списки узлов, связей и соседей данного узла:

```
G.nodes()
G.edges()
G.neighbors(3)
```

[1] Мы предлагаем вводное учебное пособие по Python в приложении A; его также можно скачать из репозитория книги на GitHub по адресу github.com/CambridgeUniversityPress/FirstCourseNetworkScience.

[2] Мы предлагаем вводное учебное пособие по библиотеке NetworkX в репозитории книги на GitHub.

И вот как выполняется перебор узлов или связей в цикле:

```
for n in G.nodes:
    print(n, G.neighbors(n))
for u,v in G.edges:
    print(u, v)
```

Мы можем создать направленную сеть («Орграф») схожим образом:

```
D = nx.DiGraph()
D.add_edge(1,2)
D.add_edge(2,1)
D.add_edges_from([(2,3),(3,4),...])
```

Обратите внимание, что связь между узлом **1** и узлом **2** отличается от связи между узлом **2** и **1**, поскольку эта сеть является направленной. Также обратите внимание, что, когда мы добавляем связь, узлы добавляются автоматически, если они еще не существуют. Это очень удобно. Существуют функции для получения размера и числа связей:

```
D.number_of_nodes()
D.number_of_edges()
```

Когда мы запрашиваем соседей узла в направленной сети, мы получаем узлы, соединенные с этим узлом непосредственно входящими и исходящими связями. Помимо этого, есть также функции для получения только тех ребер, которые соответственно ведут к этому узлу либо из этого узла, именуемые предшественниками и преемниками:

```
D.neighbors(2)
D.predecessors(2)
D.successors(2)
```

Наконец, существуют функции для генерирования сетей многих типов. Обычно этим функциям требуются аргументы, задающие число узлов или связей. Ниже приведен исходный код для генерирования нескольких сетей, показанных на рис. 1.2:

```
B = nx.complete_bipartite_graph(4,5)
C = nx.cycle_graph(4)
P = nx.path_graph(5)
S = nx.star_graph(6)
```

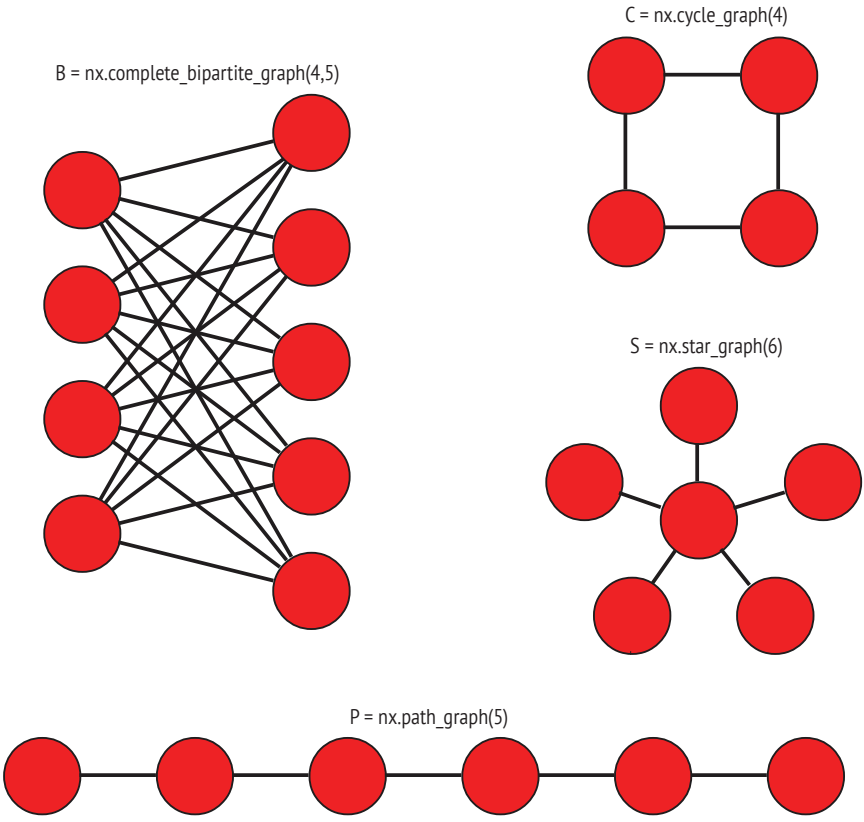


Рис. 1.2 Несколько простых сетей, сгенерированных функциями библиотеки NetworkX: полная двудольная (B), циклическая (C), звезда (S) и путь (P). Понятие *полной* сети представлено в следующем далее разделе

Мы настоятельно рекомендуем вам прочитать учебное руководство по networkx3[1] и добавить в закладки его документацию[2]. И помните, что Google и StackOverflow – это ваши друзья в ситуациях, когда вы застреваете!

1.3. Плотность и разреженность

Максимальное число связей в сети ограничено возможным числом отличимых соединений между узлами системы. Следовательно, максимальное число связей определяется числом пар узлов. Сеть с максимальным числом связей, в которой все возможные пары узлов соединены связями, называется *полной сетью*.

[1] См. networkx.github.io/documentation/stable/tutorial.html.
[2] См. networkx.github.io/documentation/stable/.

Максимальное число связей в ненаправленной сети с N узлами – это число отличимых пар узлов:

$$L_{\max} = \binom{N}{2} = \frac{N(N-1)}{2}. \quad (1.1)$$

Интуитивно каждый узел может соединяться с $N - 1$ другими узлами, и их всего N. Однако это означало бы засчитывать каждую пару дважды, поэтому мы делим на два. В направленной сети каждая пара узлов должна засчитываться дважды, по одному разу для каждого направления, поэтому $L_{\max} = N(N - 1)$. Подсчет возможных пар объектов среди множества из N объектов – это то, с чем мы снова столкнемся в книге чуть позже. У математиков есть название для формулы $\binom{N}{2}$: «из N по два».

Двудольная сеть является *полной*, если каждый узел в одной группе соединен со всеми узлами в другой группе (см. пример B на рис. 1.2). В данном случае $L_{\max} = N_1 \times N_2$, где N_1 и N_2 – это размеры двух групп.

Доля возможных связей, которые существуют фактически, одинаковая с долей пар узлов, которые соединены фактически, называется *плотностью* сети. Полная сеть имеет максимальную плотность, равную единице. Однако фактическое число связей обычно намного меньше максимального, так как большинство пар узлов напрямую друг с другом не соединены. Следовательно, плотность часто намного меньше единицы – на порядки в большинстве реально существующих крупных сетей. Этот признак является важным и помогает в работе со структурой сети. Мы называем его *разреженностью*. Интуитивно чем меньше ребер в сети, тем она разреженнее.

Плотность сети с N узлами и L связями равна:

$$d = L/L_{\max}. \quad (1.2)$$

В ненаправленной сети она задается уравнением

$$d = L/L_{\max} = \frac{L}{N(N-1)}. \quad (1.3)$$

а в направленной сети плотность равна:

$$d = L/L_{\max} = \frac{2L}{N(N-1)}. \quad (1.4)$$

В полной сети, $d = 1$ по определению, поскольку $L = L_{max}$. В разреженной сети $L \ll L_{max}$, и, следовательно, $d \ll 1$. Когда сеть становится очень крупной, мы можем наблюдать, как число связей увеличивается как функция от числа узлов. Мы говорим, что сеть разреженная, если число связей растет пропорционально числу узлов ($L \sim N$) или даже медленнее. Если вместо этого число связей растет быстрее, например квадратично вместе с размером сети ($L \sim N^2$), тогда мы говорим, что сеть – плотная.

В качестве иллюстрации важности разреженности сети давайте рассмотрим пример с Facebook. На момент написания этой книги у Facebook было около 2 млрд пользователей ($N \approx 2 \times 10^9$). Если бы эта сеть была полной, то имелось бы $L \approx 10^{18}$ связей – это число с 18 нулями, и нет никакого способа хранить столь много данных! Но, к счастью, социальные сети очень разреженные, и Facebook не является исключением. Каждый пользователь имеет в среднем 1000 друзей или меньше того, вследствие чего плотность приближенно равна $d \approx 10^{-6}$. Это все же много данных, но Facebook способна ими управлять.

В табл. 1.1 представлены базовые статистические величины, касающиеся размера и плотности сети, примеры которых приведены в главе 0[1]. Хотя эти сети сильно отличаются друг от друга, все они являются разреженными.

Таблица 1.1. Базовые статистические величины примеров сетей. Типы сетей могут быть направленными (D) и/или взвешенными (W). Когда метки нет, сеть является ненаправленной и невзвешенной. Для направленных сетей мы показываем среднюю степень-на-входе (которая совпадает со средней степенью-на-выходе)

Сеть	Тип	Узлы (N)	Связи (L)	Плотность	Средняя степень ($\langle k \rangle$)
Facebook, Северо-Западный университет		10567	488 337	0.009	92.4
IMDB, кинофильмы и кинозвезды		563443	921 160	0.000006	3.3
IMDB, кинозвезды, снимавшиеся вместе	W	252999	1 015 187	0.00003	8.0
Twitter, политика США	DW	18470	48 365	0.0001	2.6
Электронная почта компании Энрон	DW	87273	321 918	0.00004	3.7
Статьи по математике в «Википедии»	D	15220	194 103	0.0008	12.8
Интернет-маршрутизаторы		190914	607 610	0.00003	6.4
Авиационные перевозки в США		546	2781	0.02	10.2
Авиационные перевозки по всему миру		3179	18 617	0.004	11.7
Взаимодействие дрожжевых белков		1870	2277	0.001	2.4
Мозг C. elegans	DW	297	2345	0.03	7.9
Экологическая пищевая паутина Everglades	DW	69	916	0.2	13.3

Библиотека NetworkX позволяет легко измерять плотность направленных и ненаправленных сетей:

[1] Наборы данных для этих сетей доступны в репозитории книги на GitHub: github.com/CambridgeUniversityPress/FirstCourseNetworkScience.

```
nx.density(G)
nx.density(D)
CG = nx.complete_graph(8471)  # крупная полная сеть
print(nx.density(CG))         # калькулятор не нужен!
```

1.4. Подсети

Во многих случаях нас интересует подмножество сети, которое само по себе является сетью и называется *подсетью* (или *подграфом*). Подсеть получается путем отбора подмножества узлов и *всех* связей между этими узлами.

На рис. 1.3 представлено несколько иллюстраций подсетей ненаправленных и направленных сетей. Обилие определенных типов подсетей и их свойств имеет важное значение для характеризации реально существующих сетей. В качестве примера *клика* представляет собой полную подсеть: подмножество узлов, связанных друг с другом. Любая подсеть полной сети является кликой, потому что все пары узлов в сети соединены и, следовательно, все пары узлов в любой подсети соединены тоже.

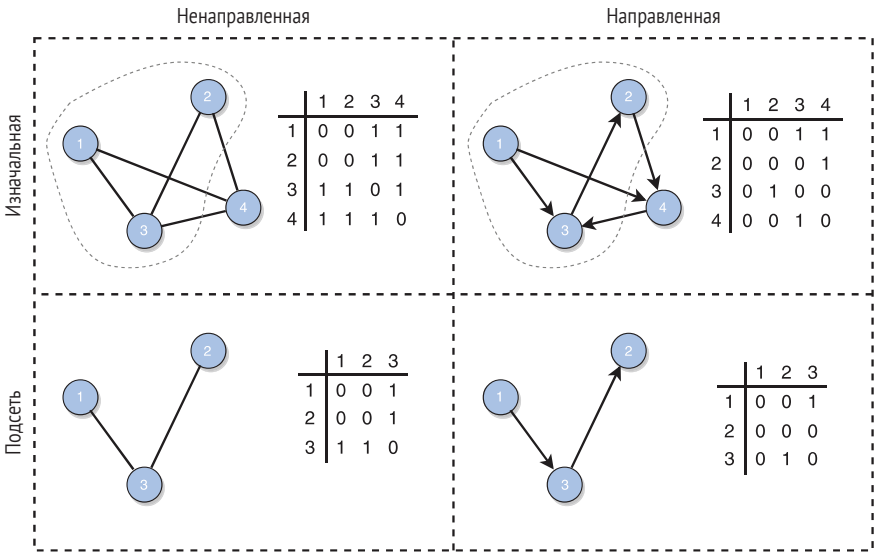


Рис. 1.3 Примеры сетей и подсетей. Мы также показываем представление каждой сети в форме матрицы смежности (см. раздел 1.9)

Особым типом подсети является *эгосеть* узла, которая представляет собой подсеть, состоящую из выбранного узла – именуемого *эго*, – и его соседей. Эгосети часто изучаются в анализе социальных сетей.

Используя библиотеку NetworkX, мы можем сгенерировать подсеть данной сети, указав подмножество узлов:

```
K5 = nx.complete_graph(5)
clique = nx.subgraph(K5, (0,1,2))
```

1.5. Степень

Степень узла – это число его связей или соседей. Мы обозначаем степень узла i через k_i. Рис. 1.4 иллюстрирует степень нескольких узлов в ненаправленной сети. Узел без соседей, например узел a на данном рисунке, имеет нулевую степень ($k = 0$) и называется *узлом-одиночкой*, или синглетоном.

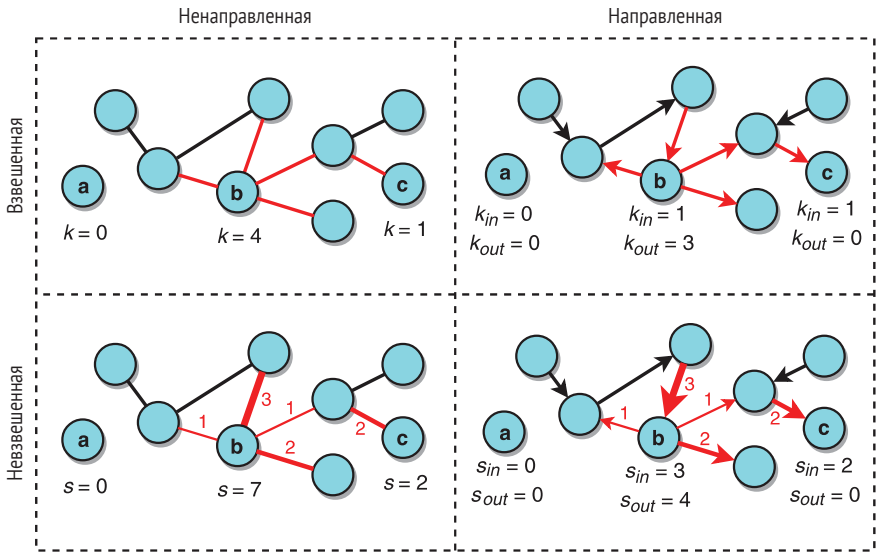


Рис. 1.4 Иллюстрации степени и силы в направленных, ненаправленных, взвешенных и невзвешенных сетях. Связи узлов **a**, **b** и **c** вместе с их весами выделены красным цветом, и показаны их степени, или силы

Средняя степень сети обозначается через $\langle k \rangle$. Это важное свойство и (прямо пропорционально) относится к ее плотности.

Средняя степень сети определяется уравнением

$$\langle k \rangle = \frac{\sum_i k_i}{N}. \qquad (1.5)$$

Поскольку каждая связь вносит свой вклад в степень двух узлов в ненаправленной сети, числитель уравнения (1.5) может записываться как $2L$. Из определения плотности для ненаправленной сети [уравнение (1.3)] $2L = dN(N-1)$. Следовательно,

$$\langle k \rangle = \frac{2L}{N} = \frac{dN(N-1)}{N} = d(N-1), \tag{1.6}$$

и наоборот

$$d = \frac{\langle k \rangle}{N-1}. \tag{1.7}$$

Это имеет смысл: максимально возможная степень узла равна $k_{max} = N - 1$ и получается, когда узел соединен с каждым другим узлом. Интуитивно плотность – это соотношение между средней и максимальной степенью.

В табл. 1.1 показана средняя степень примеров сетей, показанных в главе 0. В библиотеке NetworkX есть функция, которая возвращает степень данного узла. Без аргументов она возвращает словарь со степенью каждого узла:

```
G.degree(2)  # возвращает степень узла 2
G.degree()   # возвращает степень всех узлов сети G
```

В главе 3 мы увидим, что степени отдельных узлов сети являются очень важными свойствами для характеризации структуры сети. До сих пор мы определяли степень в ненаправленных сетях. Далее мы распространим определение на направленные и взвешенные сети.

1.6. Направленные сети

В графическом представлении сети направленная природа связей изображается посредством стрелки, указывающей направление каждой связи. Главное различие между направленными и ненаправленными сетями представлено на рис. 1.1. В ненаправленной сети наличие связи между двумя узлами соединяет смежные узлы в обоих направлениях. С другой стороны, из наличия связи в направленной сети не обязательно вытекает наличие связи в противоположном направлении. Этот факт имеет важные последствия для связности (соединенности) направленной сети, как будет подробнее рассмотрено в главе 2.

Когда мы рассматриваем степень узла в направленной сети, мы должны думать о входящих и исходящих связях отдельно. Число входящих связей, или предшественников, узла i называется *степенью-на-входе* и обозначается через k_i^{int}. Число исходящих связей, или преемников, узла i называется *степенью-на-выходе* и обозначается через k_i^{out}. На рис. 1.4 показаны степень-на-входе и степень-на-выходе нескольких узлов в направленной сети.

Мы уже определили плотность для направленной сети (уравнение (1.4)). Мы можем определить среднюю степень-на-входе и среднюю степень-на-выходе аналогично уравнению (1.5).

В библиотеке NetworkX есть функции, которые возвращают степень-на-входе и степень-на-выходе данного узла. Если сеть является направленной, то функция `degree` возвращает суммарную степень, которая представляет собой сумму степени-на-входе и на-выходе:

```
D.in_degree(4)
D.out_degree(4)
D.degree(4)
```

1.7. Взвешенные сети

В графическом представлении сети взвешенная природа связей изображается посредством отрезков разной ширины, обозначающих вес каждой связи. Нулевой вес эквивалентен отсутствию связи. Главное различие между взвешенными и невзвешенными сетями представлено на рис. 1.1.

Взвешенная сеть может быть направленной или ненаправленной; давайте сначала рассмотрим более простой случай ненаправленной взвешенной сети. Мы можем измерить степень узла в взвешенной сети, игнорируя веса. Тем не менее бывает важно учитывать веса. Поэтому мы можем определить *взвешенную степень*, или *силу* узла, как сумму весов его связей. Схожим образом мы можем определить *силу-на-входе* и *силу-на-выходе* для случая направленной взвешенной сети. Оба случая показаны на рис. 1.4.

Взвешенная степень, или *сила*, узла i в ненаправленной взвешенной сети обозначается через

$$s_i = \sum_j w_{ij}, \tag{1.8}$$

где w_{ij} – это вес связи между узлами i и j. Мы исходим из допущения, что $w_{ij} = 0$, если между i и j нет связи. Степень-на-входе и степень-на-выходе можно обобщить на силу-на-входе и на силу-на-выходе аналогичным образом в направленной взвешенной сети:

$$s_i^{in} = \sum_j w_{ji}, \tag{1.9}$$

$$s_i^{out} = \sum_j w_{ij}, \tag{1.10}$$

где w_{ij} – это вес направленной связи из i в j.

В библиотеке NetworkX как к графам, так и к орграфам могут быть прикреплены атрибуты «веса». При добавлении нескольких взвешенных связей каждая указывается как триплет, где третьим элементом является вес:

```
W = nx.Graph()
W.add_edge(1,2,weight=6)
W.add_weighted_edges_from([(2,3,3),(2,4,5)])
```

Мы можем получить список связей с ассоциированными весовыми данными, например, если нам нужно распечатать связи с крупным весом:

```
for (i,j,w) in W.edges(data='weight'):
    if w > 3:
        print('(%d, %d, %d)' % (i,j,w))  # пропустить связь (2,3)
```

Наконец, мы можем получить силу данного узла, используя функцию `degree` и указав весовой атрибут:

```
W.degree(2, weight='weight')  # сила узла 2
# равна 6 + 3 + 5 = 14
```

1.8. Многослойные и темпоральные сети

В показанной на рис. 0.7 сети авиационных перевозок в США связи представляют прямые рейсы между аэропортами независимо от того, какие конкретно авиалинии выполняют эти рейсы. Но классифици-

рование рейсов в зависимости от соответствующих им авиалиний полезно в ряде ситуаций. Мы, возможно, захотим предсказывать распространение задержек в расписании по сети авиалинии или исследовать последствия таких задержек для передвижения пассажиров. На самом деле каждая коммерческая авиалиния пытается сначала перепланировать пассажиров на свои собственные рейсы, потому что перебронировать их на рейсы другой компании дорого. Поэтому сеть авиационных перевозок конкретной авиалинии имеет свою собственную идентичность, даже несмотря на то, что она переплетена с сетями других авиалиний. В этих случаях выгодно представлять систему в виде *многослойной сети* (т. е. комбинации слоев), где каждый слой представляет сеть авиационных перевозок конкретной авиалинии: узлами являются аэропорты, связями – рейсы, выполняемые одной и той же компанией.

Если каждый слой в многослойной сети строится на одном и том же множестве узлов, то такая сеть называется *мультиплексной*. Сеть авиационных перевозок является примером мультиплекса. Еще одним примером является социальная сеть, в которой разные слои представляют разные типы социальных отношений. Например, один слой может представлять дружеские привязанности, другой слой – семейные узы, еще один – связи с коллегами и т. д. Узлы в каждом слое представляют одинаковых индивидуумов.

Темпоральная сеть – это частный случай мультиплекса. Связи являются динамическими, поскольку соответствующие взаимодействия между узлами происходят в разное время. Узлы тоже могут иметь динамический характер, поскольку они могут появляться и исчезать на разных стадиях эволюции сети. Например, сети пользовательской активности в Twitter являются темпоральными, потому что публикации, ретвиты и упоминания происходят в разное время, что можно определить по их меткам времени. Мы можем разделить временной промежуток темпоральной сети на поочередные интервалы: все узлы и связи, существующие в течение каждого интервала, составляют моментальный *снимок* системы. Каждый снимок может интерпретироваться как один слой мультиплекса, как показано на рис. 1.5.

В многослойной сети существуют *внутрислойные связи*, соединяющие пары узлов в одном слое, и *межслойные связи*, соединяющие пары узлов в разных слоях. В частном случае мультиплексных сетей межслойные связи соединяют каждый узел слоя с его противоположностью в других слоях. Такие связи называются *стыками*, потому что они состыковывают копии одного и того же узла в разных слоях.

Традиционно мультиплексные сети анализировались путем агрегирования данных из разных слоев и последующего изучения результирующей сети. Например, сети на рис. 0.3 и 0.7 представляют агрегации мультиплексных сетей, соответствующих временным интервалам и разным авиалиниям. Агрегированная сеть обычно взвешена, даже если связи мультиплекса не являются таковыми, потому

что обычно одну и ту же пару узлов в разных слоях соединяет несколько связей, которые превращаются в единую взвешенную связь в агрегированной системе. Например, связи на рис. 0.3 взвешены по числу повторных твитов одного пользователя другим. Но агрегация отбрасывает много ценной информации, предоставляемой изначальной многослойной системой. В случае авиационных перевозок слияние сетей, соответствующих разным авиакомпаниям, не позволяет нам изучать переходы пассажиров между такими сетями, что может потребоваться в случае забастовок или технических проблем, затрагивающих конкретную авиалинию.

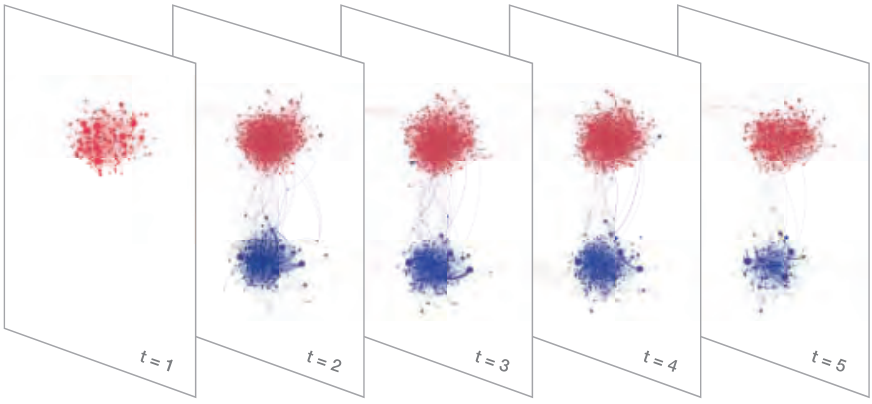


Рис. 1.5 Темпоральная сеть политических ретвитов. Каждый моментальный снимок содержит связи с ретвитами с отметками времени в определенном интервале времени. Агрегируя эти снимки во времени, мы получаем статическую сеть, показанную на рис. 0.3

В общем случае каждый слой может характеризоваться своим собственным множеством узлов и связей. Поэтому слои могут представлять совершенно разные графы, и в результате система представляет собой *сеть сетей*. Здесь межслойные связи могут представлять отношения зависимости между узлами сетей. Давайте рассмотрим электроэнергетическую сеть, которая соединяет электростанции и центры спроса через высоковольтные линии электропередачи. Станции управляются компьютерами, которые проводят мониторинг и управляют производством и передачей электроэнергии. Эти компьютеры соединены через интернет. В свою очередь интернет-маршрутизаторы зависят от электростанций в части их электроснабжения. Поэтому у нас есть система с двумя состыкованными сетями: электросетью и интернетом.

В такой состыкованной системе одна сеть может влиять на другую в целях оптимизирования доставки; по мере надобности сеть может переконфигурироваться для перенаправления электроэнергии. Однако такого рода сеть сетей также может привносить непредсказуемые уязвимости. Проблема с программно-информационным обеспечением или атака могут вывести из строя один или несколько узлов в элект-

росети, а интернет без электричества в регионе тоже может выйти из строя, что приведет к отказам в работе других узлов и в крайнем случае – катастрофическому эффекту домино, именуемому *каскадирующим сбоем*, затрагивающим крупную часть сети. По этим причинам сети сетей являются предметом интенсивного изучения.

Для упрощения в этой книге мы сосредоточимся в основном на сетях с одним типом узла и одним типом связи. В ненаправленной сети мы будем исходить из допущения, что пара узлов соединяется не более одной связью. (Если сеть является направленной, то могут существовать две связи, по одной в каждом направлении, как показано на рис. 1.1.) В дополнение к этому мы не будем рассматривать *самонаправленные циклы*, или связи, соединяющие узел с самим собой; мы будем считать, что каждая связь соединяет два отдельных узла.

1.9. Представления сетей

В целях сохранения сети в компьютерном файле или памяти и ее последующего извлечения оттуда нам нужен способ формального представления ее узлов и связей. Существует несколько возможных представлений сетей. Самое простое – это *матрица смежности*, матрица $N \times N$, в которой каждый элемент представляет связь между узлами, индексируемыми соответствующей строкой и столбцом.

> Элемент a_{ij} матрицы смежности представляет связь между узлами i и j. $a_{ij} = 1$, если i и j являются смежными, $a_{ij} = 0$ в противном случае.

На рис. 1.3 мы показываем графические иллюстрации разных ненаправленных и направленных сетей и соответствующих им матриц смежности.

В случае ненаправленных сетей матрица смежности симметрична: мы можем менять местами строки и столбцы, и матрица не меняется. Следовательно, половина матрицы содержит избыточную информацию. В случае направленных сетей матрица смежности не является симметричной. В случае невзвешенных сетей элементы принимают только значения один или ноль, чтобы обозначать соответственно наличие или отсутствие связи. В случае взвешенных сетей матричные элементы могут принимать любые значения, соответствующие весам связей. Мы уже встречались с элементами матрицы смежности для взвешенных сетей (w_{ij} в уравнениях (1.8)–(1.10)).

В библиотеке NetworkX можно получать и распечатывать матрицы смежности и использовать матричное представление для получения и задания атрибутов связи:

```
print(nx.adjacency_matrix(G))  # граф
G.edge[3][4]
G.edge[3][4]['color']='blue'
print(nx.adjacency_matrix(D))  # орграф
D.edge[3][4]
D.edge[4][3]                   # не такой же, что и приведенный выше
print(nx.adjacency_matrix(W))  # взвешенный граф
W.edge[2][3]
W.edge[2][3]['weight'] = 2
```

Хотя представление в форме матрицы смежности соответствует математическому формализму сетей, оно неэффективно для хранения реально существующих сетей, которые обычно имеют крупные размеры и разрежены. Требуемое пространство для хранения растет как квадрат размера сети (N^2), но если сеть разрежена, то бо́льшая часть этого пространства тратится впустую на хранение нулей (несуществующих связей). В крупных разреженных сетях более компактным представлением сети является *список смежности*, структура данных, в которой хранится список соседей по каждому узлу. Списки смежности эффективно представляют разреженные сети, поскольку игнорируются несуществующие связи; рассматриваются только существующие связи (ненулевые значения матрицы смежности).

Библиотека NetworkX предлагает средства для перебора сетевого списка смежности в цикле и извлечения связей и их атрибутов. Например, вот один из способов распечатать соседей каждого узла:

```
for n,neighbors in G.adjacency():
    for number,link_attributes in neighbors.items():
        print('(%d, %d)' % (n,number))
```

Третье, не менее эффективное представление сети – это *список ребер*, в котором каждая связь представлена в виде пары соединенных узлов. Нам также может потребоваться перечислить узлы отдельно в случае узлов-одиночек, которые не будут появляться ни в одной из пар. В случае взвешенных сетей каждая связь представляется в виде тройки, где третьим элементом является вес.

В этой книге для хранения сетей мы будем использовать представление в форме списка ребер. Библиотека NetworkX имеет функции для записи и чтения файлов сетей с использованием этого представления. Вы можете просмотреть формат файла списка ребер самостоятельно:

```
nx.write_edgelist(G, "file.edges")
G2 = nx.read_edgelist("file.edges")        # G2 такой же, что и G
nx.write_weighted_edgelist(W, "wf.edges")  # сохранить веса
with open("wf.edges") as f:
```

```
    for line in f:
        print(line)
W2 = nx.read_weighted_edgelist("wf.edges") # W2 такой же, что и W
```

1.10. Рисование сетей

О сети можно узнавать многое, рисуя и изучая ее графическое представление. Для этого требуется *алгоритм компоновки сети*, чтобы размещать каждый узел на плоскости. (Есть также сложные трехмерные компоновки, но в этой книге мы не обсуждаем их.) Существует целый ряд алгоритмов компоновки, служащих для представления разных типов сетей; например, для рисования сети авиационных перевозок на рис. 0.7 мы использовали *географическую компоновку*. Для относительно малых сетей компоновки размещающие узлы вдоль концентрических кругов или слоев могут выявлять важную иерархическую структуру. Наиболее популярным классом алгоритмов компоновки сети являются *алгоритмы компоновки по направлению силы* (force-directed layout algorithm), которые используются для визуализации большинства примеров сетей в главе 0. Во вставке на рис. 0.7 тоже используется компоновка по направлению силы.

Цели алгоритма компоновки по направлению силы заключаются в размещении узлов в таком ключе, чтобы соединенные узлы располагались близко друг к другу, все связи имели одинаковую длину, а число пересечений связей минимизировалось. В целях получения представления о принципе работы компоновки по направлению силы вообразите силу, которая отталкивает любые два узла друг от друга, подобно силе между двумя частицами с одинаковым электрическим зарядом. Далее вообразите пружину, соединяющую любые два связанных узла, создающую силу притяжения, когда они находятся слишком далеко друг от друга. Алгоритмы компоновки по направлению силы симулируют такую физическую систему, вследствие чего узлы движутся, минимизируя энергию системы: соединенные узлы будут двигаться навстречу друг другу и удаляться от узлов, не соединенных с ними.

Результатом является не только эстетически приятный рисунок, но и – иногда – визуализация наиболее очевидных сообществ в сети, как мы видели в главе 0. Например, поскольку на рис. 0.3 люди в сообществе (прогрессивном или консервативном) тесно соединены друг с другом, они в итоге группируются в компоновке вместе.

Библиотека NetworkX имеет функцию рисования сети, в которой используется элементарный алгоритм компоновки сети:

```
import matplotlib.pyplot
nx.draw(G)
```

Обратите внимание, что для рисования требуется интерфейс графопостроения, такой как Matplotlib. Он достаточно хорошо работает для малых сетей, имеющих, к примеру, менее 100 узлов. Для более крупных сетей существуют более совершенные инструменты визуализации. Примеры в главе 0 визуализированы с помощью алгоритма компоновки *ForceAtlas2* инструмента визуализации Gephi.

1.11. Резюме

Мы представили несколько базовых определений и величин, которые позволяют нам описывать сеть.

1. Сеть состоит из двух множеств элементов: узлов и связей, соединяющих пары узлов.
2. Подсеть – это подмножество сети, включающее несколько ее узлов и все связи между ними.
3. В направленных сетях связи имеют направление. Может существовать связь от узла **1** к узлу **2** и не обязательно из узла **2** в узел **1**. В ненаправленных сетях связи являются взаимными.
4. Во взвешенных сетях связи имеют ассоциированные веса, которые представляют атрибуты связи, такие как важность, сходство, расстояние, трафик и т. д. В невзвешенных сетях все связи одинаковы.
5. Многослойные сети имеют разные типы узлов и связей, поделенных на взаимосвязанные слои. Если в каждом слое узлы одинаковы, то такая многослойная сеть называется мультиплексной.
6. Плотность сети – это доля пар соединенных узлов. Сеть является полной, если все пары узлов соединены, вследствие чего плотность равна единице. Большинство реально существующих сетей являются разреженными, а значит они имеют очень малую плотность.
7. Степень узла – это число соседей. В направленных сетях узлы имеют степень-на-входе (in-degree) и степень-на-выходе (out-degree), измеряющие соответственно число входящих и исходящих связей. Если сеть является взвешенной, то сила узла равна сумме весов ее связей. Узлы взвешенных направленных сетей имеют силу-на-входе (in-strength) и силу-на-выходе (out-strength).
8. Списки смежности и списки ребер являются эффективными представлениями, служащими для хранения разреженных сетей.
9. NetworkX – это популярная и удобная программная библиотека для программирования сетей на языке Python.

Определения в этой главе образуют базовый словарь науки о сетях. В следующих главах будут представлены дополнительные величины и свойства, чтобы иметь возможность описывать, анализировать и моделировать реально существующие сети и узнавать то, что они говорят нам о базовых системах и явлениях.

1.12. Дальнейшее чтение

Есть несколько других отличных учебников по науке о сетях, которые выходят за рамки вводного материала этой книги. Калдарелли и Чесса (2016) окунаются чуть-чуть глубже в науку о данных нескольких тематических исследований. Если вы заинтересованы завернуть в сторону физики, то подумайте об учебнике Барабаши (2016); если вы хотите разведать связи с экономикой и социологией, то мы рекомендуем учебник Исли и Клейнберга (2010). Более продвинутым темам по физике, математике и социальным наукам посвящено много книг на выбор (Вассерман и Фауст, 1994; Кальдарелли, 2007; Баррат и соавт., 2008; Коэн и Хавлин, 2010; Боллобас, 2012; Дороговцев и Мендес, 2013; Латора и соавт., 2017; Ньюман, 2018).

Кивеля и соавт. (2014) и Боккалетти и соавт. (2014) представили влиятельные обзоры многослойных сетей. Обзор темпоральных сетей представлен Холме и Сарамяки (2012). Гао и соавт. (2012) анализируют сети сетей. Катастрофический сбой в этих сетях обсуждается Рейсом и соавт. (2014) и Радикки (2015).

Для получения справочной информации о рисовании сетей обратитесь к Ди Баттиста и соавт. (1998). Алгоритмы компоновки сети по направлению силы (также именуемые пружинной компоновкой) были введены Идесом (1984) и усовершенствованы Камадой и Каваи (1989) и Фрухтерманом и Рейнгольдом (1991). Алгоритм компоновки ForceAtlas2, используемый для многих визуализаций в этой книге, был разработан Джакоми и соавт. (2014).

Упражнения

1.1 Ознакомьтесь с учебным материалом главы 1 в репозитории книги на GitHub[1].

1.2 Рассмотрите сеть с N узлами. При наличии одной связи каково максимальное число узлов, которые может соединять связь? При наличии одного узла каково максимальное число связей, которые могут соединять с этим узлом?

1.3 Рассмотрите дорожную карту на рис. 0.9. Решетчатая структура этой сети означает, что большинство узлов имеет одинаковую степень. Какова наиболее распространенная степень узлов в этой сети?

1.4 Рассмотрите дорожную карту на рис. 0.9. На Манхэттене много улиц с односторонним движением. Это означает, что хорошая сетевая модель потока дорожного движения, вероятно, будет иметь

[1] См. github.com/CambridgeUniversityPress/FirstCourseNetworkScience.

направленные связи. Взгляните подграф этой сети с решетчатой соединенностью и в котором все улицы имеют одностороннее движение (т. е. каждый узел представляет четырехпутный перекресток двух улиц с односторонним движением). Какова наиболее распространенная степень-на-входе узлов в этом подграфе? Какова наиболее распространенная степень-на-выходе?

1.5 Какой объем сети можно использовать для представления объема дорожного движения между каждой парой смежных перекрестков на дорожной карте Манхэттена (рис. 0.9)?

1.6 Рассмотрите направленную сеть из N узлов. Теперь рассмотрите суммарную степень-на-входе (т. е. сумму степеней-на-входе по всем узлам в сети). Сравните ее с аналогичной суммарной степенью-на-выходе. Что из перечисленного ниже должно соответствовать действительности для любой такой сети?

a. Суммарная степень-на-входе должна быть меньше, чем суммарная степень-на-выходе.
b. Суммарная степень-на-входе должна быть больше, чем суммарная степень-на-выходе.
c. Суммарная степень-на-входе должна быть равна суммарной степени-на-выходе.
d. Ни один из этих пунктов не соответствует действительности во всех случаях.

1.7 Рассмотрите ретвитную сеть в Twitter, где пользователи являются узлами, и мы хотим показать, сколько раз данный пользователь ретвитнул другого пользователя. Какой тип связи лучше всего отражает эту связь?

a. Ненаправленная невзвешенная.
b. Ненаправленная взвешенная.
c. Направленная невзвешенная.
d. Направленная взвешенная.

1.8 Рассмотрите граф совместной встречаемости хештегов в Twitter. В этой сети хештеги являются узлами, и связь между двумя хештегами указывает на частоту появления этих двух хештегов в твитах вместе. Какой тип связи лучше всего отражает эту связь?

a. Ненаправленная невзвешенная.
b. Ненаправленная взвешенная.
c. Направленная невзвешенная.
d. Направленная взвешенная.

1.9 Рассмотрите сеть, созданную из персонажей истории или пьесы. Узлы – это люди, и между двумя узлами существует связь, если эти персонажи когда-либо вступают в диалог. Какой тип ребра может представлять это отношение? Обоснуйте свой ответ.

a. Ненаправленная невзвешенная.
b. Ненаправленная взвешенная.

c. Направленная невзвешенная.
d. Направленная взвешенная.

1.10 Предположим, что мы хотим создать более сложную версию диалоговой сети, которая улавливает то, сколько каждый персонаж говорит и с кем. Какой тип связи лучше всего отражает это отношение?
a. Ненаправленная невзвешенная.
b. Ненаправленная взвешенная.
c. Направленная невзвешенная.
d. Направленная взвешенная.

1.11 Представьте, что в вашей социальной сети есть подсеть, в которой вы и 24 ваших товарища (всего 25 человек) дружите друг с другом. Как называется такая подсеть? И сколько связей содержится в подсети?

1.12 Рассмотрите ненаправленную сеть с N узлами. Какое максимальное число связей может существовать в этой сети?

1.13 Рассмотрите двудольную сеть из N узлов, N_1 узлов типа 1 и N_2 узлов типа 2 (таких что $N_1 + N_2 = N$). Каково максимальное число связей в этой сети?

1.14 Имея полную сеть A с N узлами и двудольную сеть B тоже с N узлами, что из приведенного ниже соответствует действительности для любого $N > 2$?
a. Сеть A имеет больше связей, чем сеть B.
b. Сеть A имеет такое же число связей, как и сеть B.
c. Сеть A имеет меньше связей, чем сеть B.
d. Ни один из этих пунктов не соответствует действительности по всем таким $N > 2$.

1.15 Вспомните, что в полной сети существует связь между каждой парой узлов. Мы знаем, что полная ненаправленная сеть из N узлов имеет $N(N-1)/2$ ребер. Должна ли любая ненаправленная сеть из N узлов и $N(N-1)/2$ связей быть с неизбежностью полной? Объясните, почему да или почему нет.

1.16 Рассмотрите приведенную ниже матрицу смежности:

$$\begin{array}{c} \\ A \\ B \\ C \\ D \\ E \\ F \end{array} \begin{array}{c} \begin{array}{cccccc} A & B & C & D & E & F \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 2 & 1 & 3 & 1 & 1 & 0 \end{pmatrix} \end{array}. \tag{1.11}$$

Запись в *i*-й строке и *j*-м столбце указывает вес связи между узлом *i* и узлом *j*. Например, запись во второй строке и третьем столбце равна 2, а это значит, что вес связи из узла **B** в узел **C** равен 2. Какую сеть представляет эта матрица?

a. Ненаправленная невзвешенная.
б. Ненаправленная взвешенная.
c. Направленная невзвешенная.
d. Направленная взвешенная.

1.17 Рассмотрите сеть, определенную матрицей смежности в уравнении (1.11). Сколько узлов в этой сети? Сколько связей? Существуют ли какие-либо самонаправленные циклы?

1.18 Рассмотрите сеть, определенную матрицей смежности в уравнении (1.11). Существуют ли какие-либо узлы, в которых есть исходящие связи с каждым другим узлом? Если да, то какие узлы? Существуют ли какие-либо узлы, в которых есть входящие связи из каждого другого узла? Если да, то какие узлы?

1.19 Рассмотрите сеть, определенную матрицей смежности в уравнении (1.11). Приемник определяется как узел с входящими связями, но без исходящих связей. Какие узлы в сети, если таковые имеются, обладают этим свойством?

1.20 Рассмотрите сеть, определенную матрицей смежности в уравнении (1.11). Какова сила-на-входе узла **C**? Какова его сила-на-выходе?

1.21 Конвертируйте сеть, определенную матрицей смежности в уравнении (1.11), в ненаправленный невзвешенный граф. (При конвертировании направленного графа в ненаправленный узлы *i* и *j* соединяются в ненаправленном графе, если имеется направленная связь из *i* в *j* или из *j* в *i* или и та и другая.) Возможно, вам захочется распечатать полученную матрицу и/или нарисовать диаграмму сети для справки. Сколько узлов в этой конвертированной сети? Сколько связей?

1.22 Рассмотрите невзвешенную, ненаправленную версию сети, определенную матрицей смежности в уравнении (1.11), построенную, как описано в упражнении 1.21. Какова минимальная степень в этой сети? Какова максимальная степень? Какова средняя степень? Какова плотность?

1.23 Вообразите две разные ненаправленные сети, каждая с одинаковым числом узлов и связей. Должны ли обе сети иметь одинаковую максимальную и минимальную степень? Объясните, почему да или почему нет. Должны ли они иметь одинаковую среднюю степень? Объясните, почему да или почему нет.

1.24 Мы видели, что сеть Facebook невероятно разрежена. Предположим, что у нее примерно 1 млрд пользователей, у каждого из которых в среднем 1000 друзей.

- Предположим, Facebook публикует свой годовой отчет, и он показывает, что, хотя число пользователей в сети осталось прежним, среднее число друзей на одного пользователя увеличилось. Будет ли это означать, что плотность сети увеличилась, уменьшилась или осталась прежней?
- Предположим вместо этого, что как число пользователей, так и среднее число друзей на одного пользователя удвоились. Будет ли это означать, что плотность сети увеличилась, уменьшилась или осталась прежней?

1.25 Netflix хранит данные о предпочтениях клиентов, используя большую двудольную сеть, соединяющую пользователей с кинофильмами, которые они посмотрели и/или оценили. Библиотека кинофильмов Netflix содержит около 100 000 названий, если засчитывать потоковую передачу и отправку DVD по почте. В четвертом квартале 2013 года Netflix сообщила, что у нее около 33 млн пользователей. Предположим, что средняя степень пользователя в этой сети составляет 1000. Примерно сколько связей в этой сети? Считаете ли вы эту сеть разреженной или плотной? Объясните.

1.26 Netflix хранит данные о предпочтениях клиентов, используя большую двудольную сеть, соединяющую пользователей с заголовками. Предположим, что с 2013 по 2014 год библиотека Netflix осталась прежнего размера, в то время как число пользователей увеличилось. Далее предположим, что средняя степень пользователя в этой сети осталась неизменной. Увеличилась ли плотность этой сети, уменьшилась или осталась прежней?