Содержание

Об авторе

Марко Бонцанини – исследователь-аналитик из Лондона (Соединенное Королевство). Имеет докторскую степень в области информационного поиска Лондонского университета королевы Марии. Специализируется на анализе текстовой информации и поисковых приложениях и многие годы с удовольствием занимался решением разнообразных задач управления информацией и науки о данных.

Ведет персональный блог на http://marcobonzanini.com, где обсуждает различные технические темы, главным образом связанные с языком Python, анализом текстовой информации и наукой о данных.

Когда Марко не занят работой над проектами на Python, он с удовольствием принимает участие в жизни сообщества, посещая конференции и неформальные встречи PyData с разработчиками, а также очень любит варить домашнее пиво.

Эта книга является результатом долгого процесса, который выходит далеко за рамки простой подготовки материалов книги. Многие так или иначе принимали участие в формировании конечного результата. В первую очередь я хотел бы поблагодарить команду издательства Packt Publishing, в особенности Сонали Вернекар (Sonali Vernekar) и Сиддхеш Салви (Siddhesh Salvi) за предоставленную возможность работать над этой книгой и за помощь в течение всего процесса. Доктор Вейай «Уэйн» Сюй (Dr. Wei Ai «Wayne» Xu) выступил в роли научного редактора настоящей книги и внес множество предложений по ее улучшению, за что ему отдельное спасибо. Многие коллеги и друзья в случайных разговорах, глубоких обсуждениях и во время работы над проектами в прошлом немало поспособствовали повышению качества материала, представленного в этой книге. Особо хотел бы поблагодарить доктора Мигеля Мартинеса-Альвареса (Dr. Miguel Martinez-Alvarez), Марко Кампана (Marco Campana) и Стефано Кампана (Stefano Campana). Я также рад быть частью лондонского сообщества PyData, группы умнейших людей, которые регулярно встречаются, чтобы поговорить о Python и науке о данных в располагающей для этого атмосфере. И напоследок, но не в последнюю очередь, хотел бы отметить Даниэлу (Daniela), поощрявшую меня в течение всего процесса, делясь со мной своими мыслями, предлагая улучшения и обеспечивая уютную обстановку по возвращении после работы.

О рецензенте

Вейай Уэйн Сюй – доцент в отделе коммуникаций Массачусетского университета (Амхерст, США) и сотрудничающий с институтом вычислительной социологии этого же университета. Ранее Сюй преподавал теорию сетей в институте сетевых наук Бостонского северо-восточного университета. Результаты его исследований в области онлайн-сообществ, «сарафанного радио» и социального капитала публиковались в различных авторитетных научных журналах. Сюй также оказывал содействие четырем национальным проектам в области стратегической коммуникации и общественного мнения. Кроме основной работы является соучредителем лаборатории данных под названием CuriosityBits Collective (http://www.curiositybits.org/).

Предисловие

За прошедшие несколько лет популярность социальных ресурсов существенно выросла благодаря тому, что все больше и больше пользователей стало обмениваться разнообразными видами информации их посредством. Компании используют социально-медийные платформы для продвижения своих брендов, профессионалы регистрируют общедоступные учетные записи и используют их для налаживания связей, а обычные пользователи занимаются обсуждением любых тем. Увеличение числа пользователей также означает увеличение объемов данных, ожидающих анализа.

Вы, читатель этой книги, вероятнее всего разработчик, инженер, аналитик, исследователь или студент, который хотел бы применить приемы анализа к данным, хранящимся на социальных ресурсах. С этой точки зрения вы, как практик в области анализа данных (или будущий практик), не будете испытывать недостатка в потенциальных возможностях и вызовах.

Книга «Анализ социальных медиа на Python» даст основные инструменты, которые помогут вам использовать это обилие данных в своих интересах. Знакомство с главными инструментами анализа данных на Python начнется с предоставления вводной информации о методах обработки естественного языка, машинного обучения, анализа социальных сетей и визуализации данных. Последующее пошаговое руководство по самым популярным социально-медийным платформам, включая Twitter, Facebook, Google+, Stack Overflow, Blogger, YouTube и др., расскажет, как получить доступ к данным этих сетей и как применять различные виды анализа для извлечения полезной информации из необработанных данных.

В настоящей книге затрагиваются три главных аспекта, а именно:

- **социально-медийные API**: каждая платформа обеспечивает доступ к своим данным по-своему. Понимание, как с ними взаимодействовать, даст возможность ответить на вопросы: *как получить данные?* и *какие данные можно получить?* Это важно потому, что невозможно анализировать данные, не имея к ним доступа. Каждая глава посвящена отдельной социально-медийной платформе и подробно рассказывает, как взаимодействовать с соответствующим API;
- **приемы анализа данных**: простое извлечение данных из API не имеет особой ценности. Следующий шаг – ответ на вопрос: *что можно делать с данными?* Каждая глава описывает понятия, которые помогут вам понять суть того или иного вида анализа данных и в чем заключается их ценность. В теоретическом плане мы лишь слегка затронем внешние аспекты, особо не вдаваясь в подробности, которые оставим академическим учебникам. Главная цель этой книги – привести практические примеры, которые дадут вам возможность легко приступить к работе;
- **инструменты Python для анализа данных**: поняв, что можно делать с данными, нам останется ответить на последний вопрос: *как это делается?* Python зарекомендовал себя как один из главных языков для анализа

данных. Простые синтаксис и семантика языка вместе с его богатой экосистемой поддержки научных вычислений обеспечивают пологую кривую обучения для новичков и всевозможные изощренные инструменты для экспертов. Настоящая книга знакомит с главными библиотеками для Python, используемыми в мире научных вычислений, в том числе NumPy, pandas, NetworkX, scikit-learn, NLTK и многими другими. Практические примеры имеют форму коротких сценариев, которые можно использовать (и возможно расширять) для выполнения различных и интересных видов анализа данных из социальных ресурсов, к которым вы имеете доступ.

Если исследование области, охватывающей эти три главные темы, представляет интерес, тогда эта книга для вас.

О ЧЕМ РАССКАЗЫВАЕТСЯ В КНИГЕ

Глава 1 «*Социальные медиа, социальные данные и Python*» вводит главные понятия анализа данных с использованием языка Python применительно к социальным медиа. Здесь читатель найдет краткий обзор технологий машинного обучения, обработки естественного языка, анализа социальных сетей и визуализации данных, а также обсуждение главных инструментов Python для анализа данных и справочную информацию о настройке среды программирования на Python.

Глава 2 «*#Твиттер – хештеги, темы и временные ряды*» открывает практическое обсуждение анализа данных из социальной сети Twitter. После подготовки приложения, взаимодействующего с программным интерфейсом Twitter API, эта глава объясняет, как получить данные посредством потокового API и как выполнить частотный анализ хештегов и текста. Здесь также обсуждаются некоторые виды анализа временных рядов для поиска закономерностей распределения твитов во времени.

Глава 3 «*Пользователи, читатели и сообщества в Twitter*» продолжает обсуждение анализа данных из социальной сети Twitter, сосредоточив внимание на пользователях и взаимодействиях между ними. Эта глава показывает, как выявлять связи и диалоги между пользователями. Здесь объясняются такие интересные приемы, как кластеризация (сегментация) пользователей и измерение степени влияния и вовлеченности пользователей.

Глава 4 «*Сообщения, страницы и взаимодействия пользователей в Facebook*» посвящена социальной сети Facebook и ее пограммному интерфейсу Facebook Graph API. В данной главе подробно разбираются способы взаимодействия с Graph API, включая аспекты безопасности и конфиденциальности, и приводятся примеры извлечения сообщений из профиля пользователя и страниц в Facebook. Принципы анализа временных рядов и степени вовлеченности пользователей применяются к таким взаимодействиям пользователей, как комментарии, отметки «Нравится» и реакции.

Глава 5 «*Тематический анализ в Google+*» охватывает социальную сеть Google. Здесь подробно разбираются способы доступа к централизованной платформе Google и обсуждаются примеры поиска содержимого и пользователей в Google+. Эта глава также показывает, как встроить данные, поступающие из Google API, в собственное веб-приложение, сконструированное на основе микрофреймворка Flask.

Глава 6 «*Вопросы и ответы в сети Stack Exchange*» описывает порядок работы с вопросами и ответами и в качестве главного примера использует сеть Stack Exchange. Здесь читатель узнает, как искать пользователей и содержимое на различных сайтах этой сети, и прежде всего на Stack Overflow. Используя дампы данных данной сети для онлайн-обработки, эта глава знакомит с методами машинного обучения с учителем для классификации текстов и показывает, как встроить модель машинного обучения в приложение реального времени.

Глава 7 «*Блоги, RSS, Википедия и обработка естественного языка*» знакомит с приемами анализа текстовой информации. Всемирная паутина полна возможностей с точки зрения анализа текстов, и эта глава покажет, как взаимодействовать с различными источниками данных, такими как WordPress.com API, Blogger API, каналы RSS и Wikipedia API. На основе текстовых данных формализуются и расширяются основные понятия обработки естественного языка, кратко упоминаемые на протяжении всей книги. Затем читателю на специальных примерах демонстрируется процедура извлечения ссылок на сущности из произвольного текста.

Глава 8 «*Анализ других данных*» напоминает о многочисленных возможностях анализа данных, доступных за пределами наиболее распространенных социальных сетей. Демонстрирует примеры извлечения данных из YouTube, GitHub и Yelp, а также обсуждает создание собственного программного клиента, если конкретная платформа его не предоставляет.

Глава 9 «*Связанные данные и Семантическая паутина*» дает обзор Семантической паутины и связанных с ней технологий. Здесь рассматриваются темы связанных данных, микроформатов и RDF и предлагаются примеры анализа семантической информации из DBpedia и Википедии.

Что потребуется для работы с книгой

Прилагаемые к книге примеры программного кода предполагают наличие у читателя последней версии Python в Linux, macOS либо Windows. Програмный код был протестирован в Python 3.4.* и Python 3.5.*. Более старые версии (Python 3.3.* или Python 2.*) в явном виде не поддерживаются.

Глава 1 «*Социальные медиа, социальные данные и Python*» содержит инструкции по настройке локальной среды разработки и знакомит с кратким списком инструментов, используемых на протяжении всей книги. Вам также понадобятся некоторые важнейшие библиотеки для научных вычислений (NumPy, pandas и matplotlib), машинного обучения (scikit-learn), обработки естественного языка (NLTK) и анализа социальных сетей (NetworkX).

Кому адресована книга

Эта книга адресована в меру опытным разработчикам на языке Python, желающим использовать общедоступные API для сбора данных из социально-медийных платформ и выполнять статистический анализ для извлечения полезной информации из данных. Книга предполагает элементарное знакомство со стандартной библиотекой Python и приводит практические примеры, которые станут вам ориентиром для создания своего собственного аналитического проекта на основе социальных данных.

Соглашения

В этой книге используются разные стили оформления текста, которые разделяют различные виды информации. Ниже приводятся примеры этих стилей и поясняется их значение.

Элементы программного кода в тексте, имена таблиц в базах данных, имена папок и файлов, расширения файлов, пути к каталогам в файловой системе, фиктивные адреса URL, ввод пользователя и учетные записи в Twitter оформляются так: «Кроме того, атрибут `genre` представлен списком с переменным числом значений».

Блоки кода оформляются следующим образом:

```
from timeit import timeit
import numpy as np

if __name__ == '__main__':
    setup_sum = 'data = list(range(10000))'
    setup_np = 'import numpy as np;'
    setup_np += 'data_np = np.array(list(range(10000)))'
```

Когда потребуется привлечь ваше внимание к определенному фрагменту в блоке программного кода, он будет выделяться жирным:

```
Наберите ваш вопрос либо наберите "exit", чтобы выйти.
> What's up with Gandalf and Frodo lately? They haven't been in the Shire for a while...
Вопрос: What's up with Gandalf and Frodo lately? They haven't been in the Shire for a while...
Предсказанные метки: plot-explanation, the-lord-of-the-rings
```

Ввод или вывод в командной строке будет оформляться так:

```
$ pip install --upgrade [package name]
```

Новые термины и **важные слова** будут выделены жирным. Текст, отображаемый на экране, например в меню или в диалогах, будет оформляться так: «На странице конфигурации Keys and Access Tokens (Ключи и маркеры доступа) разработчик сможет найти ключ API и пароль, а также маркер доступа и секретный маркер доступа».

 Так оформляются предупреждения и важные примечания.

 Так оформляются советы и рекомендации.

 Так оформляются дополнения от переводчика к тексту оригинальной книги.

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 прямо 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 напишите название книги в теме письма.

Если есть тема, в которой вы квалифицированы, и вы 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 или www.дмк.рф в разделе «Читателям – Файлы к книгам».

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы удостовериться в качестве наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в тексте или в коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от расстройств и поможете нам улучшить последующие версии этой книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу dmkpress@gmail.com, и мы исправим это в следующих тиражах.

Скачивание цветных иллюстраций

Мы также подготовили файл PDF с цветными иллюстрациями, диаграммами и скриншотами, которые в этой книге имеют черно-белое оформление. Цветные иллюстрации помогут вам лучше понять обсуждаемые темы. Вы можете загрузить файл с иллюстрациями по адресу: https://www.packtpub.com/sites/default/files/downloads/MasteringSocialMediaMiningWithPython_ColorImages.pdf.

Нарушение авторских прав

Пиратство в Интернете по-прежнему остается насущной проблемой. Издательства «ДМК Пресс» и Packt очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в Интернете с незаконно выполненной копией любой нашей книги, пожалуйста, сообщите нам адрес копии или вебсайта, чтобы мы могли принять меры.

Пожалуйста, свяжитесь с нами по адресу электронной почты dmkpress@gmail.com со ссылкой на подозрительные материалы.

Мы высоко ценим любую помощь по защите наших авторов, помогающую нам предоставлять вам качественные материалы.

Вопросы

Вы можете присылать любые вопросы, касающиеся данной книги, по адресу dm@dmk-press.ru или questions@packtpub.com. Мы постараемся разрешить возникшие проблемы.

Комментарий переводчика

Весь материал настоящей книги приведен в соответствие с последними действующими версиями программных библиотек (время перевода книги — июль 2017 г.), дополнен свежей информацией и протестирован в среде Windows 10. Тестирование программного кода осуществлялось в Python 3.6.1.

Прилагаемый к книге адаптированный и скорректированный исходный код примеров лучше всего разместить в домашней папке пользователя (`/home/Ваши_проекты_Python` или `C:\Users\[ИМЯ_ПОЛЬЗОВАТЕЛЯ]\Ваши_проекты_Python`). Ниже приведена структура папки с прилагаемыми примерами:

`Chap01-Chap09`	Исходный код примеров на языке Python
`Дополнение`	Сценарий на Python для анализа данных из социальной сети «ВКонтакте»

Далее приведены особенности инсталляции некоторых используемых программных библиотек Python.

Особенности программного обеспечения

Библиотеки для Python обычно можно скачать из каталога библиотек Python PyPi (https://pypi.python.org/). Но имейте в виду, если предполагается использовать библиотеки SciPy и Scikit-learn в Windows, вам придется также установить библиотеку Numpy+MKL. Библиотека Numpy+MKL привязана к библиотеке Intel® Math Kernel Library и включает необходимые динамические библиотеки (DLL) в каталоге numpy.core. Загрузите соответствующие whl-файлы из репозитория (http://www.lfd.uci.edu/~gohlke/pythonlibs/) и установите (например, в 64-разрядной версии Windows и Python 3.6: `pip install numpy-1.13.0+mkl-cp36-cp36m-win_amd64.whl`). Соответствующая процедура установки описана ниже. Далее приводятся сведения о базовых библиотеках:

- **NumPy** – основная библиотека для научных вычислений на Python;
- **Matplotlib** – библиотека для работы с двумерными графиками. Требует наличия numpy и некоторых других пакетов;
- **Pandas** – инструмент для анализа структурных данных и временных рядов. Требует наличия пакета numpy и некоторых других;
- **Scikit-learn** – интегратор классических алгоритмов машинного обучения. Требует наличия `numpy+mkl`;
- **SciPy** – библиотека, используемая в математике, естественных науках и инженерном деле. Требует наличия `numpy+mkl`.

Факультативно:

- **Jupyter** – интерактивная вычислительная среда;
- **PyQt5** – библиотека инструментов для создания графического интерфейса, требуется для работы инструментальной среды программирования Spyder;
- **Spyder** – инструментальная среда программирования на Python.

Протокол установки библиотек

Среди прилагаемых примеров имеется файл `requirements.txt`, выполняющий установку всех используемых в книге библиотек в пакетном режиме одной командой:

```
pip3 install -r requirements.txt
```

Обратите внимание, что, прежде чем выполнить эту команду, необходимо перейти в каталог, где находится этот файл, например командой `cd`. Этот файл также можно найти в репозитории с примерами к книге по адресу: https://github.com/bonzanini/Book-SocialMediaMiningPython.

Если вы захотите установить более свежие версии библиотек и контролировать весь процесс установки, ниже предлагается список команд установки библиотек в том порядке, в каком они встречаются в тексте книги. В некоторых случаях библиотеки должны устанавливаться из whl-файлов, которые можно загрузить из репозитория (http://www.lfd.uci.edu/~gohlke/pythonlibs/).

```
pip3 install --upgrade pip или pip install -U pip
pip3 install virtualenv
pip3 install numpy
```

либо как whl: `pip3 install numpy-1.13.0+mkl-cp36-cp36m-win_amd64.whl`

```
pip3 install pandas
pip3 install scipy
```

либо как whl: `pip3 install scipy-0.19.0-cp36-cp36m-win_amd64.whl`

```
pip3 install Scikit-learn
```

либо как whl: `pip3 install scikit_learn-0.18.1-cp36-cp36m-win_amd64.whl`

```
pip3 install matplotlib
pip3 install nltk
pip3 install Pyro4
pip3 install gensim
pip3 install networkx
pip3 install tweepy
pip3 install folium
pip3 install facebook-sdk
pip3 install wordcloud
```

либо как whl: `pip3 install wordcloud-1.3.1-cp36-cp36m-win_amd64.whl`

```
pip3 install Pillow
pip3 install google-api-python-client
pip3 install flask
pip3 install beautifulsoup4
pip3 install py-stackexchange
pip3 install lxml
pip3 install requests
pip3 install feedparser
pip3 install wikipedia
pip3 install PyGithub
pip3 install yelp
pip3 install rdflib
pip3 install mf2py
pip3 install pykml
```

факультативно:

```
pip3 install jupyter
pip3 install pyqt5
pip3 install spyder
```

В зависимости от типа операционной системы, версии Python и версий библиотек версии whl-файлов для установки могут отличаться от приведенных выше, где показаны последние версии (по состоянию на август 2017) для 64-разрядной версии Windows и Python 3.6.1.

Перечень использованных примеров

Глава 1

```
python demo_gensim.py lord_of_the_rings.txt
```

Глава 2

```
python twitter_get_user_timeline.py marcobonzanini
python twitter_get_user_timeline.py PacktPub
python twitter_streaming.py \#RWC2015 \#RWCFinal rugby
python twitter_hashtag_frequency.py user_timeline_PacktPub.jsonl
python twitter_hashtag_stats.py user_timeline_PacktPub.jsonl
python twitter_mention_frequency.py user_timeline_PacktPub.jsonl
python twitter_term_frequency.py filename.jsonl
python twitter_time_series.py stream__RWC2015__RWCFinal_Rugby.jsonl
```

Глава 3

```
python twitter_get_home_timeline_toscreen.py
python twitter_get_home_timeline.py
python twitter_get_user_timeline.py marcobonzanini
python twitter_get_user_timeline.py PacktPub
python twitter_streaming.py \#RWC2015 \#RWCFinal rugby
python twitter_hashtag_frequency.py user_timeline_PacktPub.jsonl
python twitter_hashtag_stats.py user_timeline_PacktPub.jsonl
python twitter_mention_frequency.py user_timeline_PacktPub.jsonl
python twitter_term_frequency.py user_timeline_PacktPub.jsonl
python twitter_term_frequency_graph.py user_timeline_PacktPub.jsonl
python twitter_time_series.py stream__RWC2015__RWCFinal_Rugby.jsonl
python twitter_make_geojson.py --tweets stream__RWC2015__RWCFinal_Rugby.jsonl --geojson
rwc2015_final.geo.json
```

Глава 4

```
python twitter_get_user.py PacktPub
python twitter_get_user.py marcobonzanini
python twitter_followers_stats.py PacktPub
python twitter_followers_stats_set.py PacktPub
python twitter_followers_stats_numpy.py PacktPub
python twitter_influence.py marcobonzanini PacktPub
python twitter_cluster_users.py --filename users/marcobonzanini/followers.jsonl --k 5
--max-features 200 --max-ngram 3
python twitter_conversation.py stream__RWC2015__RWCFinal_Rugby.jsonl
python twitter_map_example.py --map example_map.htm
python twitter_map_basic.py --geojson rwc2015_final.geo.json --map rwc2015_final_tweets.html
python twitter_map_clustered.py --geojson rwc2015_final.geo.json --map rwc2015_final_tweets_
clustered.html
python facebook_my_profile.py
python facebook_get_friends.py
python facebook_get_my_posts.py
```

```
python facebook_get_my_posts_more_fields.py
python facebook_post_time_stats.py -f my_posts.jsonl
python facebook_get_page_info.py --page PacktPub
python facebook_get_page_posts.py --page PacktPub --n 500
python facebook_top_posts.py --page PacktPub
python facebook_top_posts_plot.py --page PacktPub
python facebook_posts_wordcloud.py --page PacktPub
```

Дополнение:

```
twitter_get_oldtweets_in_bulk.py
```

Глава 5

```
python gplus_search_example.py --query packt
python gplus_search_web_gui.py
python gplus_get_page_activities.py --page +packtpublishing --max-results 1000
python gplus_activities_keywords.py --file activities_+LarryPage.jsonl --keywords 5
```

Глава 6

```
python stack_search_keyword.py --tags "python;nosql" --n 10
python stack_search_user.py --name joel
python stack_search_user.py --name joel --sort creation --order asc
python stack_xml2json.py --xml movies.stackexchange_5.com --json movies.tags.jsonl
python stack_xml2json.py --xml movies.stackexchange_4.com --json movies.posts.jsonl --clean-post
python stack_classification_prepare_dataset.py --tags-file movies.tags.jsonl --posts-file
movies.posts.jsonl --output movies.questions4classification.jsonl  --min-df 10
python stack_classification_predict_tags.py --questions movies.questions4classification.jsonl
python stack_classification_predict_tags.py --questions movies.questions4classification.jsonl
--min-df 5
python stack_classification_save_model.py --questions movies.questions4classification.jsonl
--min-df 5 --output questions-svm-classifier.pickle
python stack_classification_user_input.py --model questions-svm-classifier.pickle
```

Глава 7

```
python blogs_wp_get_posts.py --domain marcobonzanini.com --output posts.marcobonzanini.com.
jsonl --posts 100
python blogs_blogger_get_posts.py --url http://googleresearch.blogspot.co.uk --posts 50
--output posts.googleresearch.jsonl
python blogs_rss_get_posts.py --rss-url http://feeds.bbci.co.uk/news/rss.xml --json rss.bbc.jsonl
python blogs_entities.py --entity London
```

Глава 8

```
python youtube_search_video_pagination.py --query python --n 50 --output videos.jsonl
python github_get_user.py --user bonzanini --get-repos
python github_search_user.py --query [здесь ваш запрос] --sort followers --order desc
python github_search_repos.py --query python --sort stars --order desc
python yelp_search_business.py --location London --search breakfast --limit 5
python yelp_search_business.py --location "San Francisco" --search "craft beer" --limit 5
```

Глава 9

```
python rdf_summarize_entity.py --entity "Python"
python rdf_summarize_entity.py --entity "Python_(programming_language)"
```

```
python micro_geo_wiki.py --url "https://en.wikipedia.org/wiki/London"
python micro_geo_wiki.py --url "https://en.wikipedia.org/wiki/List_of_United_States_cities_by_
population"
python micro_geo2kml.py --url "https://en.wikipedia.org/wiki/List_of_United_States_cities_by_
population" --output us_cities.kml --n 20
```

Установка библиотек Python из whl-файлов

Библиотеки для Python можно разрабатывать не только на чистом Python. Часто библиотеки пишутся на C (динамические библиотеки), и для них пишется обертка Python, или же библиотека пишется на Python, но для оптимизации узких мест часть кода пишется на C. Такие библиотеки получаются очень быстрыми, однако программисту на Python тяжелее установить их ввиду банального отсутствия соответствующих знаний или необходимых компонентов и настроек в рабочей среде (в особенности в Windows). Для решения описанных проблем разработан специальный формат (файлы с расширением `.whl`) для распространения библиотек, который содержит заранее скомпилированную версию библиотеки со всеми ее зависимостями. Формат whl поддерживается всеми основными платформами (Mac OS X, Linux, Windows).

Установка производится с помощью диспетчера библиотек `pip`. В отличие от обычной установки командой `pip install <имя_библиотеки>`, вместо имени библиотеки указывается путь к whl-файлу: `pip install <путь/к/whl_файлу>`. Например:

```
pip install C:\temp\scipy-0.19.0-cp36-cp36m-win_amd64.whl
```

Откройте окно командой строки и, выполнив команду `cd`, перейдите в каталог, где находится whl-файл. Затем просто скопируйте в команду выше имя whl-файла. В этом случае полный путь указывать не понадобится. Например:

```
pip install scipy-0.19.0-cp36-cp36m-win_amd64.whl
```

При выборе библиотеки важно, чтобы разрядности устанавливаемой библиотеки и интерпретатора совпадали. Пользователи Windows могут загрузить whl-файлы на веб-странице http://www.lfd.uci.edu/~gohlke/pythonlibs/ Кристофа Голька из Лаборатории динамики флуоресценции Калифорнийского университета в г. Ирвайн. Библиотеки там постоянно обновляются, и в архиве содержится все, что только может понадобиться.

Установка и настройка инструментальной среды Spyder

Spyder — это инструментальная среда для научных вычислений на языке Python (Scientific PYthon Development EnviRonment) для Windows, Mac OS X и Linux. Это простая, легковесная и бесплатная интерактивная среда разработки на Python, которая предлагает функционал, аналогичный среде разработки на MATLAB, включая готовые к использованию виджеты `PyQt5` и `PySide`: редактор исходного кода, редактор массивов данных `NumPy`, редактор словарей, консоли Python и IPython и многое другое.

Чтобы установить среду Spyder в Ubuntu Linux, используя системный диспетчер пакетов, достаточно выполнить всего одну команду:

```
sudo apt-get install spyder
```

Чтобы установить ее с использованием диспетчера библиотек `pip`:

```
sudo apt-get install python-qt5 python-sphinx
sudo pip install spyder
```

А чтобы обновить:

```
sudo pip install -U spyder
```

Установить Spyder в Fedora 25 можно командой:

```
dnf install python-spyder
```

Установка среды Spyder в Windows:

```
pip install spyder
```

Среда Spyder требует обязательной установки библиотеки `PyQt5`.

Глава 1

Социальные медиа, социальные данные и Python

Настоящая книга посвящена технологиям *анализа данных* из *социальных сетей* с использованием языка *Python*. Три выделенных ключевых термина в предыдущем предложении помогают определить целевую аудиторию настоящей книги: любой разработчик, инженер, аналитик, исследователь или студент, интересующийся исследованием области, где встречаются эти три темы.

Эта глава охватывает следующие темы:

- социальные медиа и социальные данные;
- процесс анализа данных из социальных сетей;
- настройка среды Python для разработки приложений;
- инструменты Python для обработки научных данных;
- обработка данных на Python.

Начало

Во втором квартале 2015 г. компания Facebook объявила, что число активных пользователей этой социальной сети превысило 1,5 миллиарда. В 2013 г. компания Twitter заявила, что ежедневно обрабатывает более 500 миллионов твитов. Читателям также будет интересно узнать, что в 2015 г. веб-сайт Stack Overflow объявил о том, что с момента его открытия было задано больше 10 миллионов вопросов.

Данные числа являются лишь верхушкой айсберга, учитывая экспоненциональный рост популярности социальных медиа по мере увеличения числа пользователей, обменивающихся информацией посредством различных платформ. Это обилие данных дает практикам анализа данных уникальные возможности. Цель настоящей книги состоит в том, чтобы помочь читателю овладеть приемами использования API социальных медиа для сбора данных, которые можно проанализировать при помощи инструментов Python, и получить интересные выводы о том, как пользователи взаимодействуют между собой.

Настоящая глава закладывает основы для начального обсуждения проблем и возможностей анализа данных из социальных медиа и знакомит с некоторыми инструментами Python, которые будут использоваться в последующих главах.

Социальные медиа – проблемы и возможности

В традиционных средствах массовой информации пользователи, как правило, являются простыми потребителями. Информация течет в одном направлении: от публикатора к пользователям. Социальные медиа разрушают эту модель, позволяя каждому пользователю одновременно быть потребителем и публикатором. По этой теме написано большое количество академических статей, в которых авторы пытались определить, что на самом деле означает термин «*социальные медиа*», например Андреас М. Каплан (Andreas M. Kaplan) и Майкл Хэенлейн (Michael Haenlein) в своей статье «*Users of the world, unite! The challenges and opportunities of Social Media*» («Пользователи всех стран, объединяйтесь! Проблемы и возможности социальных медиа»). Разные социально-медийные платформы объединяют следующие аспекты:

- интернет-приложения;
- информация, генерируемая пользователями;
- сетевые взаимодействия.

По своей сути социальные медиа – это интернет-приложения. Ясно, что прогресс в области интернет- и мобильных технологий способствовал распространению социальных медиа. Посредством мобильного телефона можно мгновенно соединиться с социально-медийной платформой, опубликовать свой материал или разузнать последние новости.

Социально-медийные платформы управляются информацией, генерируемой пользователями. В противоположность традиционной модели, в них потенциальным публикатором является каждый пользователь. И, что более важно, любой пользователь может взаимодействовать с любым другим пользователем, делясь сведениями, оставляя комментарии или выражая положительную оценку посредством кнопки Like (Нравится) (то есть «лайкая», «плюсуя» или щелкая на пиктограмме с большим пальцем вверх).

Социальные медиа основаны на сетевых взаимодействиях. Как описано выше, суть социальных медиа заключается во взаимодействии пользователей между собой. Быть на связи – вот центральная идея большей части социально-медийных платформ, и информация, которую вы получаете посредством своей *ленты новостей*, обусловлена вашими связями.

С учетом этих ключевых особенностей, характерных для большинства платформ, социальные медиа используются, чтобы:

- оставаться на связи с друзьями и семьей (например, посредством Facebook);
- вести микроблог и узнавать последние новости (например, посредством Twitter);
- оставаться на связи с профессиональной сетью (например, посредством LinkedIn);
- делиться мультимедийным материалом (например, посредством Instagram, YouTube, Vimeo и Flickr);
- находить ответы на вопросы (например, посредством Stack Overflow, Stack Exchange и Quora);
- находить и организовывать интересующую информацию (например, посредством Pinterest).

Цель этой книги – дать ответ на центральный вопрос: как извлекать полезные знания из данных, поступающих из социальных медиа? А теперь отступим на шаг назад и определим, что такое *знание* и что понимать под *полезностью*.

Традиционные определения понятия «знание» пришли из информатики. Обычно «знание» изображается как часть пирамиды, иногда называемой иерархией знаний, в основании которой лежат данные, на среднем уровне – информация и наверху – знание. Эта пирамида изображена на рис. 1.1.

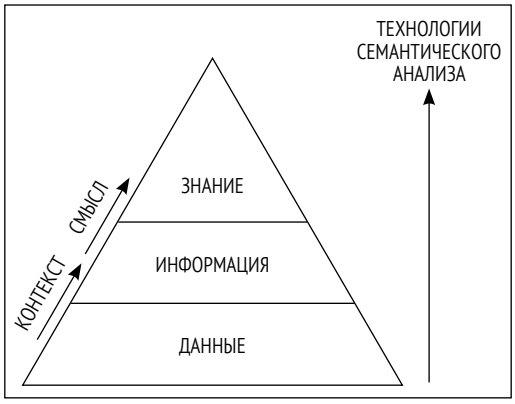

Рис. 1.1 ❖ От исходных данных к семантическому знанию

Восхождение на пирамиду отражает процесс извлечения знания из исходных данных. Путь от исходных данных к чистому знанию лежит через интеграцию контекста и смысла. По мере восхождения вверх по пирамиде применение технологий анализа помогает получать все более глубокое понимание исходных данных и, что еще более важно, понимание пользователей, которые производят эти данные. Другими словами, увеличивается полезность знаний.

В данном контексте полезное знание означает *эффективное* знание, то есть знание, которое позволяет лицу, принимающему решения, реализовать бизнес-стратегию. Прочтя эту книгу, вы поймете ключевые принципы извлечения ценности из социальных данных. Понимание, как пользователи взаимодействуют посредством социально-медийных платформ, является одним из ключевых аспектов в этом процессе.

Следующие ниже разделы формулируют некоторые проблемы и возможности анализа данных из социально-медийных платформ.

Возможности

Ключевой потенциал систем анализа данных кроется в *извлечении полезных выводов* из данных. Цель данного процесса состоит в том, чтобы ответить на интересные (а иногда и трудные) вопросы, используя технологии анализа данных, и достичь более полного понимания конкретной предметной области. Например, розничный онлайн-магазин может применить анализ данных, чтобы понять, как их потребители выбирают товары. По результатам анализа они смогут рекомен-

довать клиентам товары в зависимости от их покупательских привычек (например, пользователи, покупающие изделие A, также покупают изделие B). В общем случае это улучшает качество обслуживания клиентов и их удовлетворенность, что в итоге поможет нарастить продажи.

Многие организации в разных сферах деятельности могут применять технологии анализа данных для совершенствования своего бизнеса. Вот только некоторые примеры:

- банковское дело:
  - выявление лояльных клиентов с целью предложения им эксклюзивных программ;
  - распознавание мошеннических схем с целью уменьшения затрат;
- медицина:
  - изучение поведения пациентов с целью предсказания посещений хирурга;
  - помощь врачам в идентификации успешных способов лечения в зависимости от анамнеза пациента;
- розничная торговля:
  - изучение схем поведения во время покупок с целью улучшения качества обслуживания клиентов;
  - увеличение эффективности рекламных кампаний за счет более точной направленности;
  - анализ оперативных данных транспортных потоков для поиска самого быстрого маршрута доставки продуктов питания.

Как все это транслируется в область социальных медиа? По сути дела, вопрос состоит в том, как пользователи делятся своими данными посредством социально-медийных платформ. Организации больше не ограничиваются анализом данных, собираемых непосредственно, поскольку теперь они обладают доступом к гораздо большему объему данных.

Решение задачи сбора таких данных реализуется посредством удобных и универсальных API. Социально-медийные платформы применяют широко распространенную практику, предлагая разработчикам веб-ориентированные прикладные программные интерфейсы (Web API) для встраивания функций платформы в их приложения.

Прикладной программный интерфейс. Прикладной программный интерфейс (Application Programming Interface, API), или просто программный интерфейс, – это комплекс определений процедур и протоколов, описывающих поведение программного компонента, такого как библиотека или удаленная служба, в терминах разрешенных действий, входных и выходных данных. Используя сторонние API, разработчикам не нужно беспокоиться о внутреннем устройстве компонентов, а только о том, как его использовать.

Термином Web API мы обозначаем веб-службу, которая предоставляет ряд URI-адресов (возможно, после прохождения аутентификации) для доступа к данным. Для разработки таких API широко используется архитектурный подход, который называется **передачей состояния представления** (Representational State Transfer, REST). Программный интерфейс, реализующий архитектуру REST, называется **RESTful API**. Однако мы предпочитаем использовать более общий термин «Web API», потому что многие из существующих API не всегда строго соответствуют