Грант С. Ингерсолл
Томас С. Мортон
Эндрю Л. Фэррис

Обработка неструктурированных текстов

Поиск, организация и манипулирование

ДМК ПРЕСС ИЗДАТЕЛЬСТВО
MANNING

Грант С. Ингерсолл, Томас С. Мортон, Эндрю Л. Фэррис

ОБРАБОТКА НЕСТРУКТУРИРОВАННЫХ ТЕКСТОВ

Поиск, организация и манипулирование

Taming Text

HOW TO FIND, ORGANIZE, AND MANIPULATE IT

GRANT S. INGERSOLL
THOMAS S. MORTON
ANDREW L. FARRIS

MANNING
SHELTER ISLAND

Обработка неструктурированных текстов

ПОИСК, ОРГАНИЗАЦИЯ И МАНИПУЛИРОВАНИЕ

ГРАНТ С. ИНГЕРСОЛЛ
ТОМАС С. МОРТОН
ЭНДРЮ Л. ФЭРРИС

2-е издание, электронное

ДМК издательство
Москва, 2023

УДК 004.738.52
ББК 32.972.53
И59

Ингерсолл, Грант С.
И59 Обработка неструктурированных текстов. Поиск, организация и манипулирование / Г. С. Ингерсолл, Т. С. Мортон, Э. Л. Фэррис ; пер. с англ. А. А. Слинкина. — 2-е изд., эл. — 1 файл pdf : 416 с. — Москва : ДМК Пресс, 2023. — Систем. требования: Adobe Reader XI либо Adobe Digital Editions 4.5 ; экран 10". — Текст : электронный.

ISBN 978-5-89818-308-0

В книге описаны инструменты и методы обработки неструктурированных текстов. Прочитав ее, вы научитесь пользоваться полнотекстовым поиском, распознавать имена собственные, производить кластеризацию, пометку, извлечение информации и автореферирование. Знакомство с фундаментальными принципами сопровождается изучением реальных применений.

Издание предназначено для читателей без подготовки в области математической статистики и обработки естественных языков. Примеры написаны на Java, но сами идеи могут быть реализованы на любом языке программирования.

УДК 004.738.52
ББК 32.972.53

Электронное издание на основе печатного издания: Обработка неструктурированных текстов. Поиск, организация и манипулирование / Г. С. Ингерсолл, Т. С. Мортон, Э. Л. Фэррис ; пер. с англ. А. А. Слинкина. — Москва : ДМК Пресс, 2015. — 414 с. — ISBN 978-5-97060-144-0. — Текст : непосредственный.

ISBN 978-5-89818-308-0

ОГЛАВЛЕНИЕ

ПРЕДИСЛОВИЕ

Во времена, когда спрос на высококачественные средства обработки текста растет экспоненциально, трудно назвать хотя бы одну отрасль экономики, которая не зависела бы от той или иной текстовой информации. А в связи с развитием веб-экономики эта зависимость только усиливается. И вместе с ней быстро возрастает потребность в талантливых технических специалистах. Вот в таких условиях выходит на свет отличная, практически ориентированная книга «Обработка неструктурированных текстов», в которой вы найдете проверенные на реальном опыте рекомендации и инструкции.

Грант Ингерсолл и Дрю Фэррис, два блистательных и в высшей степени квалифицированных инженера-программиста, с которыми я работала много лет, и Тим Мортон, внесший немалый вклад в обработку естественного языка (natural language processing, NLP), предлагают прагматическое руководство тем, кто хотел бы войти в избранный круг специалистов по обработке текстов,

Грант, Дрю и Том выбрали подход, который я называю «обучение на практике ради практики», и сумели сорвать покров тайны с действительно очень сложных процессов. Для этого они не пошли по длинному пути – теоретическому семестровому курсу по NLP, а сосредоточились на существующих инструментах, реализованных до конца примерах и хорошо протестированном коде.

Для инженера-программиста этих основ будет достаточно, чтобы открыть дверь в мир примеров и упоминаемых проектов с открытым исходным кодом. И гораздо быстрее, чем вам кажется, вы превратитесь в настоящего эксперта, готового к решению реальных задач.

Лиз Лидди
Декан, ISchool,
Сиракузский Университет

ВСТУПЛЕНИЕ

Жизнь полна удивительных сюрпризов, и некоторые из них оказали определяющее влияние на мою карьеру. Было это в конце 1990 годов, я тогда был молодым программистом, занимался моделированием распределения электромагнитных полей и случайно наткнулся на предложение места разработчика в небольшой компании в городе Сиракузы, штат Нью-Йорк, которая называлась TextWise. Прочитав описание работы, я подумал, что едва ли подойду, но решил все-таки попробовать и отправил резюме. Непонятно почему, меня взяли, и так началась моя карьера в области обработки естественных языков. Кто бы мог подумать, что спустя столько лет я так и буду заниматься поиском и NLP и даже напишу книгу на эту тему.

А тогда моей первой задачей стало участие в разработке межъязыковой информационно-поисковой системы (CLIR), которая позволяла пользователю вводить запросы на английском языке, а находить и автоматически переводить документы на французском, испанском или китайском. Оглядываясь назад, я понимаю, что в первой же системе, над которой я работал, встретились все те трудные проблемы работы с текстом, которые я впоследствии так полюбил: поиск, классификация, извлечение информации, машинный перевод, а также специфические правила конкретных языков, способные свести с ума любого, кто изучает грамматику. После этого проекта я работал над самыми разными системами поиска и обработки естественных языков – от классификаторов на основе правил до вопросно-ответных систем. Позже, в 2004 году, уже на новой работе в Центре обработки естественных языков я столкнулся с Apache Lucene, поисковой системой с открытым исходным кодом, которая в те дни являлась стандартом де факто. И снова мне пришлось разрабатывать CLIR-систему, только теперь для английского и арабского языков. Поскольку мне потребовались кое-какие функции, которых в Lucene не было, я начал писать дополнения и исправлять ошибки. Спустя какое-то время я стал отправлять плоды своих трудов в репозиторий исходного кода. И пошло-поехало. Я пристрастился к проектам с открытым исходным

кодом, начав с системы машинного обучения Apache Mahout вместе с Изабель Дрост (Isabel Drost) и Карлом Веттином (Karl Wettin), а потом стал сооснователем компании Lucid Imagination, которая специализировалась на задачах поиска и анализа текстов с применением Apache Lucene и Solr.

Описав полный круг, я пришел к выводу, что поиск и NLP принадлежат к числу вопросов, определяющих предмет информатики, поскольку требуют изощренных подходов как к структурам данных, так и к алгоритмам решения задач. Добавьте сюда требования к масштабируемости, необходимой для обработки гигантских объемов данных, порождаемых пользователями веб вообще и социальных сетей в частности, – и вот вам мечта любого разработчика. Эта книга призвана заполнить пустующую на тот момент рыночную нишу – текст, написанный инженерами для инженеров и посвященный, прежде всего, использованию существующих, проверенных практикой библиотек с открытым исходным кодом для решения трудных задач обработки текста. Надеюсь, что она поможет вам в повседневной работе, а также откроет мир текстовых данных – богатейшую возможность для изучения нового.

Грант Ингерсолл

Я подпал под очарование искусственного интеллекта на втором курсе вуза, а на старших курсах решил остаться в аспирантуре и сосредоточиться на обработке естественных языков. В Пенсильванском университете я очень много узнал об обработке текстов, машинном обучении, а также об алгоритмах и структурах данных вообще. У меня также была возможность работать с некоторыми из лучших специалистов в области обработки естественных языков и набираться у них ума-разума.

В аспирантуре я занимался различными NLP-системами и принимал участие в ряде финансируемых агентством DARPA исследований по кореференции, свертыванию и порождению ответов на вопросы. В ходе этой работы я познакомился с системой Lucene и движением за ПО с открытым исходным кодом в целом. Я также обратил внимание на пробел в открытых программах обработки текстов, заполнение которого могло бы обеспечить эффективную сквозную обработку. Работая над диссертацией, я активно участвовал в проекте OpenNLP, а также продолжал изучать NLP-системы, разрабатывая систему автоматизированной оценки сочинений и кратких ответов в службе образовательного тестирования (Educational Testing Services).

Тесное сотрудничество с разработчиками ПО с открытым исходным кодом научило меня коллективной работе и позволило усовершенствоваться в своей профессии. Сейчас я работаю в компании Comcast Corporation с командами программистов, которые применяют многие описанные в этой книге приемы и инструменты. Надеюсь, эта книга станет мостом между напряженно ищущими исследователями типа тех, у кого я учился в аспирантуре, и инженерами-практиками, цель которых – использовать обработку текстов для решения реальных задач в интересах обычных людей.

Томас Мортон

Я, как и Грант, получил первое представление об информационном поиске и обработке естественных языков под руководством д-ра Элизабет Лидди, Вуджина Пайка (Woojin Paik) и прочих сотрудников компании TextWise в середине 1990-х годов. В то время TextWise была в стадии превращения из исследовательской группы в новообразованную компанию, специализирующуюся на разработке приложений на основе полученных результатов в области обработки текста. Я работал в компании много лет и все это время занимался самообразованием, открывал для себя что-то новое и общался с выдающимися людьми, которые, не убоявшись трудностей, решили научить машины понимать различные аспекты человеческого языка.

Лично я подхожу к проблеме анализа текста, прежде всего, с точки зрения разработчика программного обеспечения. Мне повезло работать с блестящими учеными и превращать их идеи из экспериментов сначала в функционирующие прототипы, а затем и в массивно масштабируемые системы. По ходу дела у меня была возможность плотно заниматься тем, что теперь принято называть наукой о данных, и я глубоко и навсегда полюбил исследование больших наборов данных и методы извлечения информации из них.

Невозможно переоценить то огромное влияние, которое открытое ПО оказало на мою карьеру. Наличие под рукой исходного кода как подспорья в исследованиях, – невероятно эффективный способ изучения новых методов и подходов к анализу текста и к разработке ПО вообще. Я приветствую всякого, кто приложил усилия, чтобы поделиться своими знаниями и опытом с другими людьми, страстно желающими сотрудничать и учиться. И особо я хочу поблагодарить отличных ребят из фонда Apache Software Foundation, неустанно взращивающих динамичную экосистему, которая способствует раз-

работке открытого ПО и помогает организовывать процессы и сплачивать людей, это ПО поддерживающих.

Инструменты и методы, описанные в этой книге, своими корнями уходят глубоко в сообщество разработчиков ПО с открытым исходным кодом. Lucene, Solr, Mahout и OpenNLP – все эти проекты выросли под опекой Apache. В этой книге мы лишь скользнули по поверхности того, что умеют эти инструменты. Нашей целью было продемонстрировать базовые концепции, лежащие в основе обработки текстов, и заложить прочный фундамент под будущие исследования в этой области.

Успехов в кодировании!

Дрю Фэррис

БЛАГОДАРНОСТИ

Работа над книгой заняла немало времени, и в ней принимало участие множество людей, которым мы с радостью выражаем свою признательность. Мы благодарны:

- пользователям и разработчикам Apache Solr, Lucene, Mahout, OpenNLP и других инструментов, упоминаемых в этой книге;
- издательству Manning Publications, не бросившему нас, а в особенности Дугласу Пандику (Douglas Pundick), Карен Тегтмейер (Karen Tegtmeyer) и Брайану Бейсу (Marjan Bace);
- Джеффу Блейелю (Jeff Bleiel), нашему выпускающему редактору, который всю дорогу подгонял нас, невзирая на наш безумный график, всегда имел наготове похвалу и сумел превратить разработчиков в авторов;
- наших рецензентов за вопросы, комментарии и критические замечания, благодаря которым книга стала лучше: Адама Тейси (Adam Tacy), Амоса Бэннистера (Amos Bannister), Клинта Ховарта (Clint Howarth), Костантино Чербо (Costantino Cerbo), Давида Вайсса (Dawid Weiss), Дениса Куриленко (Denis Kurilenko), Дуга Уоррена (Doug Warren), Фрэнка Джаниа (Frank Jania), Ганна Бирнера (Gann Bierner), Джеймса Хатуэя (James Hatheway), Джеймса Уоррена (James Warren), Джейсона Ренни (Jason Rennie), Джеффри Коупленда (Jeffrey Copeland), Джоша Рида (Josh Reed), Жульена Ниоша (Julien Nioche), Кита Кима (Keith Kim), Маниша Катьяла (Manish Katyal), Маргрит Бруггеман (Margriet Bruggeman), Массимо Перга (Massimo Perga), Никандера Бруггемана (Nikander Bruggeman), Филиппа К. Дженерта (Philipp K. Janert), Рика Вагнера (Rick Wagner), Роби Сена (Robi Sen), Саншета Диге (Sanchet Dighe), Шимона Чойнацки (Szymon Chojnacki), Тима Поттера (Tim Potter), Вайджаната Рао (Vaijanath Rao) и Джеффа Годлшрафе (Jeff Goldschrafe);
- наших соавторов, которые внесли вклад в отдельные разделы книги: Дж. Нила Рихтера (J. Neal Richter), Маниша Катьяла,

Роба Зинкова (Rob Zinkov), Шимона Чойнацки, Тима Поттера и Вайджаната Рао;
- Стивена Роуи (Steven Rowe) за скрупулезное техническое редактирование, а также за многие часы, которые он отдал написанию приложений для обработки текста в компаниях TextWise, CNLP, а также в проекте Lucene;
- д-ра Лиз Лидди за то, что она ввела Дрю и Гранта в мир анализа текстов и познакомила с таящимися в нем удивительными возможностями, а также за написание предисловия к этой книге;
- всех читателей предварительной версии книги за терпение и отзывы;
- а прежде всего, наши семьи, друзей и коллег за ободрение, моральную поддержку и понимание на протяжении всего периода, когда мы были вынуждены отдавать часть своей жизни работе над книгой.

Грант Ингерсолл

Благодарю всех своих коллег по компаниям TextWise и CNLP, которые столь много рассказали мне об анализе текстов; г-на Урдала, который привил мне интерес к математике, и г-жу Реймонд, благодаря которой я стал хорошим студентом и человеком; своих родителей, Флойда и Делорес, и детей, Джеки и Уильяма (всегда вас люблю); свою жену Робин, которая мирилась с моими занятиями допоздна и упущенными выходными – спасибо, что ты сумела вытерпеть все это!

Том Мортон

Благодарю своих соавторов за тяжкий труд и дружеские отношения; свою жену Туи и дочь Хлою за терпение, поддержку и отданное мне время; своих родственников, Мортонов и Транов, за постоянное ободрение; коллег из Пенсильванского университета и компании Comcast за поддержку и сотрудничество, а особенно На-Рай Хан (Na-Rae Han), Джейсона Балбриджа (Jason Baldridge), Ганна Бирнера (Gann Bierner) и Марту Палмер (Martha Palmer); Йорна Котманна (Jцrn Kottmann) за неустанную работу над проектом OpenNLP.

Дрю Фэррис

Благодарю Гранта за привлечение меня к этому и многим другим интересным проектам; своих коллег, прошлых и настоящих, от которых я многому научился и с которыми делил и делю страсть к ана-

лизу текстов, машинному обучению и разработке удивительных программ; свою жену Кристин и детей, Фебу, Одри и Оуэна, за терпение и поддержку, невзирая на то, что я тратил столько времени на это и другие технические предприятия – в ущерб им; всей моей семье за ободрение и проявленный интерес, а особенно маму, которой не суждено посмотреть на эту книгу в завершенном виде.

ОБ ЭТОЙ КНИГЕ

«Обработка неструктурированных текстов» – это книга о создании программных приложений, ценность которых состоит, главным образом, в использовании и манипулировании содержимым обычных письменных текстов. Это не теоретический трактат по поиску, обработке естественного языка и машинному обучению, хотя все эти вопросы обсуждаются довольно подробно. Мы стремились избегать специальной терминологии и сложной математики, а сосредоточиться на концепциях и примерах, необходимых современным программистам, архитекторам и пользователям для реализации интеллектуальных приложений обработки текста нового поколения. Кроме того, наша твердая позиция – демонстрировать примеры из реальной практики с помощью бесплатных, широко распространенных инструментов с открытым исходным кодом – Apache Solr, Mahout, OpenNLP и других.

Предполагаемая аудитория

Будет ли эта книга полезна вам? Возможно. Мы ориентировались на программистов-практиков, не имеющих солидной теоретической подготовки в проблемах поиска, обработки естественного языка и машинного обучения. На самом деле, книга рассчитана на людей, с которыми мы встречались во многих компаниях: команду разработчиков, которой поручено добавить поиск и другие средства в уже существующее приложение при том, что мало кто из них (а то и вовсе никто) не имеет опыта работы с текстом. Им необходимо хорошее введение в суть предмета, не отягощенное ненужными деталями.

Часто мы отсылаем читателя к легко доступным источникам типа википедии и фундаментальным научным статьям, тем самым подготавливая стартовую площадку для, кто хотел бы изучить предмет более подробно. И еще – хотя большинство инструментов и примеров написаны на Java, сами идеи легко переносятся на многие другие языки программирования, поэтому пишущие на Ruby, Python или еще каком-то языке тоже получат пользу от чтения этой книги.

Эта книга определенно не подойдет тем, кому интересны объяснения математических основ описываемых систем, или тем, кто жаждет академической строгости изложения, хотя, как нам кажется, она пригодится студентам, когда перед ними встанет задача реализовать идеи, почерпнутые из лекций и других книг академической направленности.

Не рассчитана эта книга и на опытных специалистов-практиков, которые за свою карьеру написали не одно приложение для обработки текстов, хотя и они смогут найти в ней подсказку-другую о том, как работать с описываемыми пакетами с открытым исходным кодом. Не раз мы слышали от практиков, что эта книга очень помогает, когда нужно быстро обучить новых членов команды концепциям, относящимся к созданию приложений для обработки текстов.

В общем и целом, мы надеемся, что эта книга станет актуальным пособием для современного программиста, пособием, которого всем нам так не хватало, когда мы только начинали свою карьеру в области обработки текста.

Структура книги

В главе 1 объясняется, почему задача обработки текста важна и в чем ее трудность. Мы дадим предварительный обзор вопросно-ответной фактографической системы, подготовив сцену для «приручения» текста с применением открытых библиотек.

В главе 2 описываются основные элементы обработки текста: лексический анализ, разбиение на блоки, грамматический разбор и частеречная разметка. Затем мы поговорим о том, как извлекать текст из файлов в распространенных форматах с помощью проекта Apache Tika с открытым исходным кодом.

Глава 3 посвящена теории поиска и основам векторной модели. Мы познакомимся с поисковым сервером Apache Solr и покажем, как с его помощью индексировать документы. Вы узнаете о количественных и качественных оценках работы поисковой системы.

В главе 4 рассматривается неточный поиск в строке с помощью префиксов и *n*-грамм. Мы рассмотрим две характеристики близости строк – меру Жаккарда и расстояние Джаро-Винклера – и объясним, как с помощью Solr находить и ранжировать соответствия.

В главе 5 представлены основные концепции распознавания именованных сущностей. Мы покажем, как находить именованные сущности с помощью проекта OpenNLP, и обсудим некоторые аспекты его функционирования. Мы также рассмотрим вопрос о на-

стройке OpenNLP на распознавание сущностей новой предметной области.

Глава 6 посвящена кластеризации текста. Из нее вы узнаете об основах стандартных алгоритмов кластеризации текстов и увидите, как кластеризация может повысить качество приложения. Мы также объясним, как производить кластеризацию целых наборов документов с помощью Apache Mahout и как кластеризовать результаты поиска с помощью Carrot[2].

В главе 7 обсуждаются основы классификации, категоризации и грамматической разметки. Мы покажем, как категоризация применяется в приложениях для обработки текста и как можно построить, обучить и использовать классификатор с помощью открытых инструментов. Мы также воспользуемся реализацией алгоритма наивной байесовской фильтрации в проекте Mahout для построения категоризатора документов.

Графические выделения и загрузка исходного кода

В этой книге много примеров кода. Код выделяется моноширинным шрифтом, чтобы было проще отличить его от обычного текста. Элементы программы, например имена методов, классов и т. д., также выделяются моноширинным шрифтом.

Многие листинги сопровождаются аннотациями, иллюстрирующими основные идеи, и пронумерованными маркерами, на которые даются ссылки в последующих пояснениях.

Многие приведенные в книге примеры довольно близки к тем, что можно найти в сети. Но для краткости мы иногда удаляли некоторые части, например комментарии, чтобы код поместился на странице.

Исходный код к этой книге можно скачать с сайта издательства по адресу www.manning.com/TamingText.

Автор в сети

Приобретение книги «Обработка неструктурированных текстов» открывает бесплатный доступ к закрытому форуму, организованному издательством Manning Publications, где вы можете оставить свои комментарии к книге, задать технические вопросы и получить помощь от автора и других пользователей. Получить доступ к форуму и подписаться на список рассылки можно на странице www.manning.com/TamingText. Там же написано, как зайти на форум после регис-

трации, на какую помощь можно рассчитывать, и изложены правила поведения в форуме.

Издательство Manning обязуется предоставлять читателям площадку для общения с другими читателями и автором. Однако это не означает, что автор обязан как-то участвовать в обсуждениях; его присутствие на форуме остается чисто добровольным (и не оплачивается). Мы советуем задавать автору какие-нибудь хитроумные вопросы, чтобы его интерес к форуму не угасал!

Форум автора в сети и архивы будут доступны на сайте издательства до тех пор, пока книга не перестанет печататься.

ОБ ИЛЛЮСТРАЦИИ НА ОБЛОЖКЕ

Рисунок на обложке книги называется «Le Marchand» – купец, или лавочник. Он взят из изданного в 19-ом веке во Франции четырехтомного собрания местных костюмов, опубликованного Сильвеном Марешалем. Все иллюстрации в нем превосходно нарисованы и раскрашены вручную. Богатое разнообразие собрания Марешаля показывает, как сильно города и области различались в культурном отношении каких-то 200 лет назад. Отделенные друг от друга, люди говорили на разных языках и диалектах. Встретив человека на улице города или в деревне, по его одежде легко было определить, откуда он родом и чем занимается.

Манера одеваться с тех пор сильно изменилась, и различия между областями, когда-то столь разительные, сгладились. Теперь трудно отличить друг от друга даже выходцев с разных континентов, что уж говорить о деревеньках или городках, разделенных несколькими километрами. Мы обменяли культурное разнообразие на иное устройство личной жизни – основанное на многостороннем и стремительном технологическом развитии.

Во времена, когда одну книгу по компьютерам трудно отличить от другой, издательство Manning откликается на новации и инициативы в компьютерной отрасли обложками своих книг, на которых представлено широкое разнообразие местных укладов быта в позапрошлом веке. Мы возвращаем его в том виде, в каком оно запечатлено на рисунках из собрания Марешаля.

ГЛАВА 1. Готовимся к приручению текста

В этой главе:

- Почему так важна задача обработки текста.
- В чем сложность обработки текста.
- Подготовка к использованию библиотек с открытым исходным кодом для обработки текста.

Раз вы читаете эту книгу, то, наверное, вы программист или, по крайней мере, подвизаетесь в области информационных технологий. Вы без особых проблем работаете с электронной почтой, системами мгновенного обмена сообщениями, Google, YouTube, Facebook, Twitter, блогами и большинством других технологий, определяющих облик нашей цифровой эпохи. Но поздравив себя с собственным техническим мастерством, задумайтесь о своих пользователях. Зачастую они испытывают муки от одного лишь объема получаемой электронной почты. Они изо всех тщатся как-то организовать данные, чтобы не утонуть в них. И, возможно, они не знают и знать не хотят о всяких там RSS или JSON, не говоря уже о поисковых системах, байесовских классификаторах или нейронных сетях. Они хотят получать ответы на свои вопросы, не просеивая многие страницы результатов. Они хотят, чтобы их почта была организована и упорядочена по важности, но при этом не желают тратить на это собственное время. И вообще, пользователям нужны инструменты, которые позволяют сосредоточиться на своей жизни и работе, а не на том, как они устроены. Они хотят контролировать – приручить – дикого зверя, каким является *текст*. Но что значит «приручить текст»? Об этом мы и будем гово-

рить далее, а пока скажем, что приручение текста подразумевает три основных вещи:

- умение находить ответы и подкрепляющее их содержимое, удовлетворяющие информационные потребности;
- умение организовывать текст (ставить пометки, делать извлечения, составлять реферат) и манипулировать им почти или совсем без вмешательства пользователя;
- умение решать обе вышеуказанные задачи в условиях постоянного роста объема входной информации.

Это подводит нас к основной цели настоящей книги: предоставить вам, программисту, средства и практические рекомендации для создания приложений, которые позволят людям лучше справляться с приливной волной коммуникаций, грозящей затопить их жизнь. Другая цель этой книги – показать, как использовать существующие, бесплатные, высококачественные библиотеки и инструменты с открытым исходным кодом.

Но прежде чем вплотную заняться декларированными целями, отступим на шаг и посмотрим, из чего состоит обработка текста и почему эта задача так трудна, а также обратимся к некоторым примерам, на которых будем иллюстрировать материал в последующих главах. Конкретно, мы дадим предварительный обзор приложения, которое будет построено к концу книги: фактографической вопросно-ответной системы. Имея это в виду, приведем обоснования для обработки текста, рассмотрев размер и форму информационного мира, в котором мы живем.

1.1. Почему так важна задача обработки текста

Забавы ради попробуйте представить день, в течение которого вы не прочли ни единого слова. Да-да, именно так: целый день без новостей, вывесок, веб-сайтов и даже телевизора.

Думаете, получилось бы? Вряд ли, разве что вы целый день проспите. А теперь подумайте, что предшествовало чтению всего этого добра: годы учебы в школе, практическое освоение в ходе общения с родителями, учителями и сверстниками, бесконечные диктанты, уроки грамматики, изложения, не говоря уже о сотнях тысяч долларов, которые тратятся на обучение одного человека в колледже. А теперь остановитесь и подумайте, сколько всего вы действительно читаете за день.

Для начала попробуйте ответить на следующие вопросы.

- Сколько вы сегодня получили сообщений по электронной почте (личных и рабочих, включая спам)?
- Сколько из них вы прочли?
- На сколько ответили немедленно? В течение часа? Дня? Недели?
- Как вы находили старое почтовое сообщение?
- Сколько блогов вы сегодня прочли?
- Сколько новостных сайтов вы посетили?
- Общались ли вы с друзьями или коллегами в чатах, Twitter или Facebook?
- Сколько раз вы искали информацию с помощью Яндекса, Google, Yahoo! или Bing?
- Сколько электронных документов вы прочли на своем компьютере? В каком формате они были (Word, PDF, простой текст)?
- Как часто вы искали что-то локально (на собственной машине или в корпоративной сети Интранет)?
- Сколько нового содержимого вы создали в виде почтовых сообщений, отчетов и т. д.?

И наконец, главный вопрос: сколько времени вы на все это потратили? Если вы типичный информационный работник, то, скорее всего, к вам относятся следующие данные, опубликованные корпорацией IDC (International Data Corporation) по результатам исследований, проведенных в 2009 году (Feldman [2009]):

> Работа с электронной почтой отнимает в среднем 13 часов в неделю... Но электронная почта уже не является единственным средством коммуникации. Социальные сети, системы мгновенного обмена сообщениями, Yammer, Twitter, Facebook и LinkedIn – вот новые каналы общения, которые могут красть рабочее время у информационного работника. В этом году среднее время, затраченное на поиск информации, составило 8,8 часов в неделю, т. е. в расчете на одного работника 14 209 долларов в год. На анализ информации уходит еще 8,1 часов, что обходится организации в 13 078 долларов ежегодно. Таким образом, эти две задачи являются очевидными кандидатами на автоматизацию. Раз уж работники тратят треть времени на поиск информации и еще четверть на ее анализ, то эта деятельность должна быть сделана максимально продуктивной.

В этом исследовании даже не учитывается, сколько времени те же самые работники тратят на создание нового содержимого в свобод-

ное от работы время. На самом деле, по оценке компании eMarketer, средний пользователь Интернета проводит в сети 18 часов в неделю; для сравнения отмечается, что на просмотр телевизора, который по-прежнему занимает доминирующее место среди развлечений, уходит 30 часов в неделю.

Так что писаное слово – будь то чтение электронной почты, поиск в Google, чтение книг или просмотр Facebook – присутствует в нашей жизни повсеместно.

Мы познакомились с той частью картины создания содержимого, которая связана с отдельными лицами. А как обстоит дело с содержимым, порождаемым коллективно? По данным IDC (за 2011 год), в 2011 году в мире было создано *1,9 зетабайт* цифровой информации, а «к 2020 году мир произведет в 50 раз больше [этой величины]». Естественно, такие прогнозы зачастую оказываются заниженными, поскольку мы не можем предсказать очередную возникшую ниоткуда тенденцию, из-за которой объем содержимого превысит ожидания.

Пусть даже добрая часть этих данных приходится на сигнальные данные, изображения, аудио и видео, все равно нужно обеспечить возможность находить их в сети, и сейчас для этой цели применяются такие подходы, как составление аналитических отчетов, добавление ключевых слов-меток и текстовых описаний, или преобразование аудиоданных в текст с помощью средств распознавания речи или ручного добавления титров. Иными словами, любая добавленная нами структурная информация оказывается текстом, который помогает понять смысл содержимого и сделать его общедоступным. Как видите, объем содержимого ошеломляет сам по себе, и это вдобавок к тому, что, как мы убедимся в следующем разделе, обработка текста является сложной задачей даже в небольшом масштабе. А пока полезно было бы подумать о том, что должны уметь идеальные приложения или инструменты, дабы обуздать наступающий на нас шквал текстовой информации. Для многих это умение быстро и эффективно отвечать на вопросы, а не просто выдавать перечень возможных ответов, который мы потом должны просматривать вручную. Более того, само задание вопроса не должно сопровождаться громоздкими ритуалами; мы хотим просто писать или произносить слова, не затрудняя себя всякими кавычками, операторами И/ИЛИ и другими вещами, которые упрощают работу машине, но усложняют человеку.

И хотя мы знаем, что наш мир не идеален, один из многообещающих подходов к приручению текста, популяризируемый программой

IBM Watson для игры в Jeopardy![1] и приложением Siri от компании Apple, – это вопросно-ответная система, способная обрабатывать вопросы на естественных языках, например английском, и возвращать *настоящие* ответы, а не просто список *возможных* ответов. В этой книге мы собираемся заложить фундамент для построения такой системы. Для этого подумаем, как она могла бы выглядеть, а затем рассмотрим простой код, который умеет находить и извлекать из текста полезную информацию; впоследствии этот код найдет применение в нашей вопросно-ответной системе. И завершим мы эту главу мыслями о том, почему построение подобной системы, равно как и других приложений для обработки естественного языка, оказывается такой трудной задачей. А, кроме того, опишем, как в последующих главах будет возводиться здание вопросно-ответной системы, а заодно и других систем для работы с текстом.

1.2. Предварительный обзор фактографической вопросно-ответной системы

С точки зрения этой книги, вопросно-ответная система (ВО-система) должна обрабатывать набор документов, который теоретически мог бы содержать ответы на интересующие пользователя вопросы. Например, источником ответов может быть википедия или подборка научно-исследовательских статей. Иными словами, предлагаемая нами ВО-система основана на выявлении и анализе текста, из которого можно было бы получить ответ, опираясь на то, что она видела в прошлом. Она не сможет выводить ответ из многих разнообразных источников. Например, если спросить систему «Кто является дядей Боба?» и если в наборе имеется документ, содержащий фразы «Отцом Боба является Ола. Братом Ола является Пауль.», то система не сможет сделать вывод, что дядей Боба является Пауль. Но если имеется фраза, в которой прямо утверждается, что «дядей Боба является Пауль», то мы ожидаем, что система сможет ответить на вопрос. Сказанное не означает, что первая задача неразрешима, просто она выходит за рамки этой книги.

Схема простой ВО-системы, описанной выше, приведена на рис. 1.1.

Разумеется, на этой простой схеме не показаны многочисленные детали и не отражена подача на вход документов, однако основные

[1] Российский аналог – «Своя игра». – *Прим. перев.*

компоненты обработки вопросов пользователей все же представлены. Во-первых, чтобы разобрать вопрос и определить, что спрашивается, обычно требуется такая базовая функциональность, как выделение слов, а равно способность понять, ответ какого типа подходит для данного вопроса. Например, ответом на вопрос «Кто является дядей Боба?», вероятно, должно быть имя человека, а на вопрос «Где находится Буффало?» – название места. Во-вторых, для поиска потенциальных ответов обычно нужно уметь быстро находить фразы, предложения или фрагменты, содержащие потенциальные ответы, не заставляя систему разбирать большие куски текста.

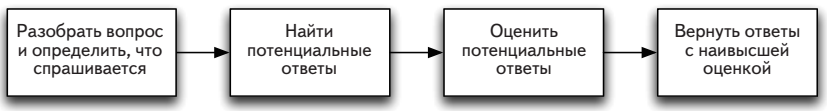

Рис. 1.1. Порядок обработки информации для получения ответов на вопросы, задаваемые простой ВО-системе

Для оценки также нужны многие базовые функции, например выделение слов, а также более глубокое понимание того, содержит ли потенциальный ответ необходимые компоненты, например упоминание человека или места. Кое-что из вышеупомянутого кажется тривиальным, если принять во внимание, с какой легкостью это делает – а точнее *думает*, что делает, – человек, однако при ближайшем рассмотрении все оказывается совсем не так просто. Памятуя об этом, рассмотрим пример обработки блока текста для поиска фрагментов и выявления разных интересных вещей, например, имен.

1.2.1. Здравствуй, доктор Франкенштейн

В свете обсуждения нашей вопросно-ответной системы и трех основных задач при работе с текстом рассмотрим пример простой операции обработки текста. Естественно, нам понадобится какой-нибудь текст. Для этой цели мы решили взять классический роман Мэри Шелли «Франкенштейн». Почему именно он? Помимо того, что он нравится авторам с литературной точки зрения, это первая книга, на которую мы наткнулись на сайте проекта Гутенберг (http://www.gutenberg.org/). К тому же, это простой текст с приличным форматированием (что для текстов, встречающихся нам в повседневной жизни, – большая редкость). Дополнительное преимущество – отсутствие копирайта и возможность бесплатно загрузить со страницы http://www.gutenberg.org/cache/epub/84/pg84.txt.