Information Retrieval

Implementing and Evaluating Search Engines

Stefan Büttcher
Google Inc.

Charles L. A. Clarke
University of Waterloo

Gordon V. Cormack
University of Waterloo

The MIT Press
Cambridge, Massachusetts
London, England

© 2010 Massachusetts Institute of Technology.

All rights reserved. No part of this book may be reproduced in any form by any electronic or mechanical means (including photocopying, recording, or information storage and retrieval) without permission in writing from the publisher.

For information about special quantity discounts, please email special_sales@mitpress.mit.edu

Typeset by the authors using LaTeX.

Printed and bound in the United States of America.

Library of Congress Cataloging-in-Publication Data
Büttcher, Stefan.
Information retrieval : implementing and evaluating search engines / Stefan Büttcher, Charles L.A. Clarke, and Gordon V. Cormack.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-262-02651-2 (hardcover : alk. paper)
1. Search engines—Programming. 2. Information retrieval. I. Clarke, Charles L. A., 1964–. II. Cormack, Gordon V. III. Title.
TK5105.884.B98 2010
025.5'24—dc22 2009048455

10 9 8 7 6 5 4 3 2 1

Preface

Information retrieval forms the foundation for modern search engines. In this textbook we provide an introduction to information retrieval targeted at graduate students and working professionals in computer science, computer engineering, and software engineering. The selection of topics was chosen to appeal to a wide audience by providing a broad foundation for future studies, with coverage of core topics in algorithms, data structures, indexing, retrieval, and evaluation. Consideration is given to the special characteristics of established and emerging application environments, including Web search engines, parallel systems, and XML retrieval.

We aim for a balance between theory and practice that leans slightly toward the side of practice, emphasizing implementation and experimentation. Whenever possible, the methods presented in the book are compared and validated experimentally. Each chapter includes exercises and student projects. Wumpus, a multi-user open-source information retrieval system written by one of the co-authors, provides model implementations and a basis for student work. Wumpus is available at `www.wumpus-search.org`.

Organization of the Book

The book is organized into five parts, with a modular structure. Part I provides introductory material. Parts II to IV each focus on one of our major topic areas: indexing, retrieval, and evaluation. After reading Part I, each of these parts may be read independently of the others. The material in Part V is devoted to specific application areas, building on material in the previous parts.

Part I covers the basics of information retrieval. Chapter 1 discusses foundational concepts including IR system architecture, terminology, characteristics of text, document formats, term distributions, language models, and test collections. Chapter 2 covers the fundamentals of our three major topics: indexing, retrieval, and evaluation. Each of these three topics is expanded upon in its own separate part of the book (Parts II to IV). The chapter provides a foundation that allows each topic to be treated more or less independently. The final chapter of Part I, Chapter 3, continues some of the topics introduced in Chapter 1, bookending Chapter 2. It covers problems associated with specific natural (i.e., human) languages, particularly *tokenization* — the process of converting a document into a sequence of terms for indexing and retrieval. An IR system must be able to handle documents written in a mixture of natural languages, and the chapter discusses the important characteristics of several major languages from this perspective.

Part II is devoted to the creation, access, and maintenance of inverted indices. Chapter 4 examines algorithms for building and accessing *static* indices, which are appropriate for document collections that change infrequently, where time is available to rebuild the index from scratch when changes do occur. Index access and query processing are discussed in Chapter 5. The chapter introduces a lightweight approach to handling document structure and applies this approach to support Boolean constraints. Chapter 6 covers index compression. Chapter 7 presents algorithms for maintaining *dynamic* collections, in which updates are frequent relative to the number of queries and updates must be applied quickly.

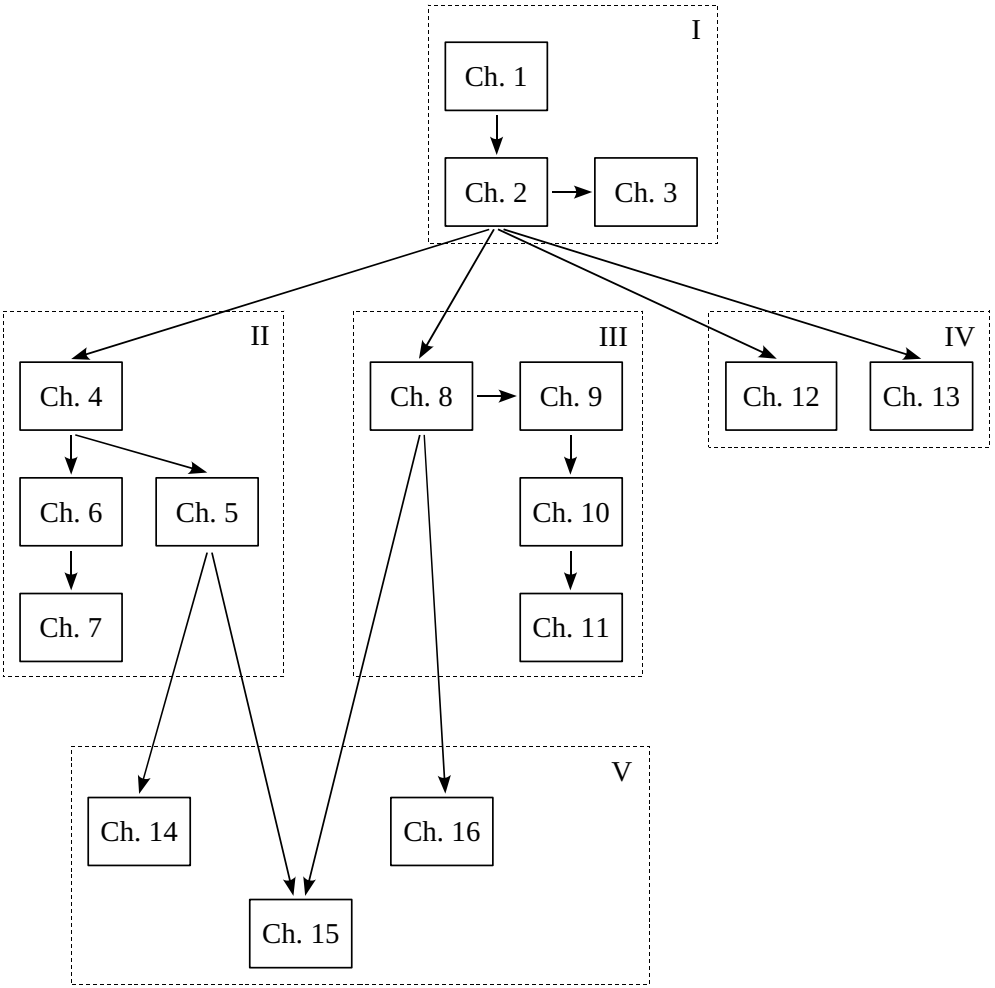
Part III covers retrieval methods and algorithms. Chapters 8 and 9 introduce and compare two major approaches to ranked retrieval based on document content: the probabilistic and language modeling approaches. The effectiveness of these approaches may be improved upon by considering explicit relevance information, by exploiting document structure, and through feedback and query expansion. We discuss the details with respect to each approach. Chapter 10 introduces techniques for document categorization and filtering, including basic machine learning algorithms for classification. Chapter 11 introduces techniques for combining evidence and parameter tuning, along with metalearning algorithms and their application to ranking.

IR evaluation forms the topic of Part IV, with separate chapters devoted to effectiveness and efficiency. Chapter 12 presents basic effectiveness measures, explores the statistical foundations for evaluating effectiveness, and discusses some recent measures, proposed over the past decade, that extend beyond the traditional IR evaluation methodology. Chapter 13 develops a methodology for the evaluation of IR system performance in terms of response time and throughput.

The chapters in Part V, the final part of the book, cover a small number of specific application areas, drawing upon and extending the more general material in one or more of the first four parts. The architecture and operation of parallel search engines is covered in Chapter 14. Chapter 15 discusses topics specific to Web search engines, including link analysis, crawling, and duplicate detection. Chapter 16 covers information retrieval over collections of XML documents.

Each chapter of the book concludes with a section providing references for further reading and with a small set of exercises. The exercises are generally intended to test and extend the concepts introduced in the chapter. Some require only a few minutes with pencil and paper; others represent substantial programming projects. These references and exercises also provide us with an opportunity to mention important concepts and topics that could not covered in the main body of the chapter.

The diagram on the next page shows the relationships between the parts and chapters of this book. Arrows indicate dependencies between chapters. The organization of the book allows its readers to focus on different aspects of the subject. A course taught from a database systems implementation perspective might cover Chapters 1–2, 4–7, 13, and 14. A more traditional information retrieval course, with a focus on IR theory, might cover Chapters 1–3, 8–12, and 16. A course on the basics of Web retrieval might cover Chapters 1–2, 4–5, 8, and 13–15. Each of these sequences represents one-half to two-thirds of the book, and could be completed in a single three-to-four month graduate course.



Organization of the book. Arrows between the individual chapters indicate dependencies.

Background

We assume that the reader possesses basic background knowledge consistent with an undergraduate degree in computer science, computer engineering, software engineering, or a related discipline. This background should include familiarity with: (1) basic data structuring concepts, such as linked data structures, B-trees, and hash functions; (2) analysis of algorithms and time complexity; (3) operating systems, disk devices, memory management, and file systems. In addition, we assume some fluency with elementary probability theory and statistics, including such concepts as random variables, distributions, and probability mass functions.

Acknowledgments

A number of our colleagues took the time to review drafts of individual chapters related to their areas of expertise. We particularly thank Eugene Agichtein, Alina Alt, Lauren Griffith, Don Metzler, Tor Myklebust, Fabrizio Silvestri, Mark Smucker, Torsten Suel, Andrew Trotman, Olga Vechtomova, William Webber, and Justin Zobel for their many valuable comments. We also thank our anonymous referees for their positive reviews and feedback.

Several classes of graduate students were subjected to early drafts of this material. We thank them for their patience and tolerance. Four students — Mohamad Hasan Ahmadi, John Akinyemi, Chandra Prakash Jethani, and Andrew Kane — read drafts with great care, helping to identify and fix many problems. Three other students — Azin Ashkan, Maheedhar Kolla, and Ian MacKinnon — volunteered to run our first attempt at an in-class evaluation effort in the fall of 2007, thus contributing to many of the exercises in Part I. Jack Wang proofread the material on CJK languages in Chapter 3. Kelly Itakura provided input on the Japanese language.

Web site

The authors maintain a Web site of material related to the book, including errata and links to cited papers, at `ir.uwaterloo.ca/book`.

Notation

For convenient reference, the list below summarizes notation that occurs frequently in the book. Additional notation is introduced as needed.

$\mathcal{C}$	a text collection
d	a document
$E[X]$	the expected value of the random variable X
$f_{t,d}$	the number of occurrences of the term t within the document d
l_{avg}	the average length of all documents in the collection
$l_{\mathcal{C}}$	the size of the collection $\mathcal{C}$, measured in tokens
l_d	the length of the document d, measured in tokens
l_t	the length of t's postings list (i.e., number of occurrences)
$\mathcal{M}$	a probability distribution; usually a language model or a compression model
N	the number of documents in the collection
N_t	the number of documents that contain the term t
n_r	the number of relevant documents
$n_{t,r}$	the number of relevant documents that contain the term t
$\Pr[x]$	the probability of the event x
$\Pr[x \mid y]$	the conditional probability of x, given y
q	a query
q_t	the number of times term t appears in the query q
t	a term
$\mathcal{V}$	the vocabulary of a text collection
$\vec{x}$	a vector
$\lvert\vec{x}\rvert$	the length of the vector $\vec{x}$
$\lvert\mathcal{X}\rvert$	the cardinality of the set $\mathcal{X}$