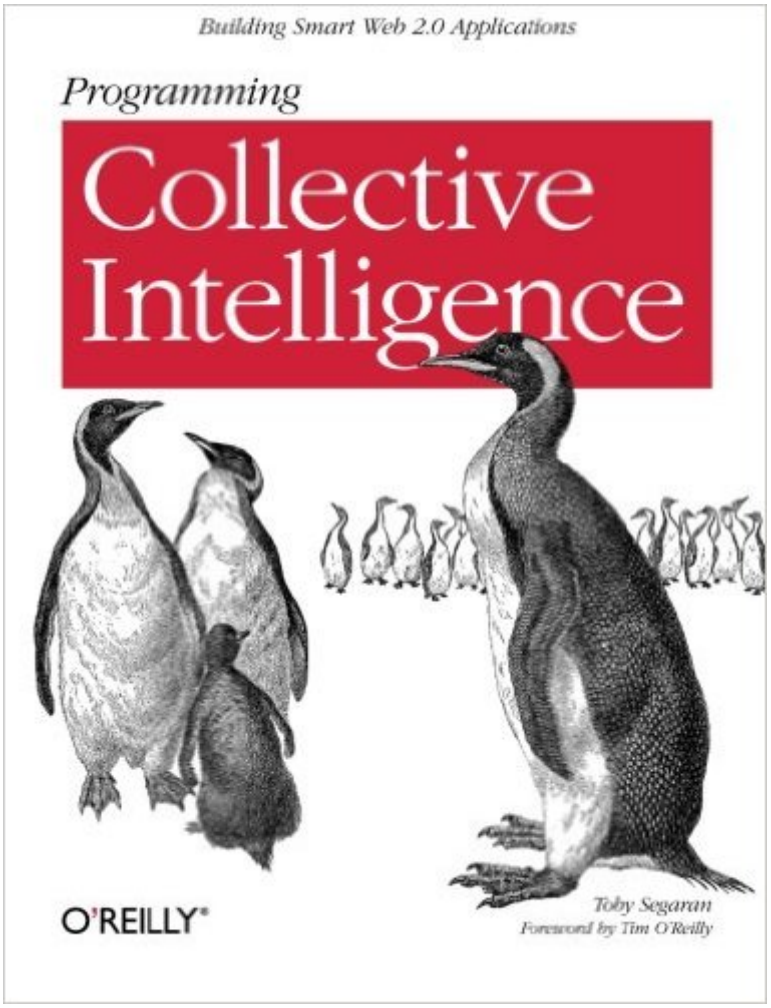
Programming Collective Intelligence: Building Smart Web 2.0 Applications



Synopsis

Want to tap the power behind search rankings, product recommendations, social bookmarking, and online matchmaking? This fascinating book demonstrates how you can build Web 2.0 applications to mine the enormous amount of data created by people on the Internet. With the sophisticated algorithms in this book, you can write smart programs to access interesting datasets from other web sites, collect data from users of your own applications, and analyze and understand the data once you've found it. Programming Collective Intelligence takes you into the world of machine learning and statistics, and explains how to draw conclusions about user experience, marketing, personal tastes, and human behavior in general--all from information that you and others collect every day. Each algorithm is described clearly and concisely with code that can immediately be used on your web site, blog, Wiki, or specialized application. This book explains: Collaborative filtering techniques that enable online retailers to recommend products or media Methods of clustering to detect groups of similar items in a large dataset Search engine features--crawlers, indexers, query engines, and the PageRank algorithm Optimization algorithms that search millions of possible solutions to a problem and choose the best one Bayesian filtering, used in spam filters for classifying documents based on word types and other features Using decision trees not only to make predictions, but to model the way decisions are made Predicting numerical values rather than classifications to build price models Support vector machines to match people in online dating sites Non-negative matrix factorization to find the independent features in adataset Evolving intelligence for problem solving--how a computer develops its skill by improving its own code the more it plays a gameÂ Each chapter includes exercises for extending the algorithms to make them more powerful. Go beyond simple database-backed applications and put the wealth of Internet data to work for you. "Bravo! I cannot think of a better way for a developer to first learn these algorithms and methods, nor can I think of a better way for me (an old AI dog) to reinvigorate my knowledge of the details." -- Dan Russell, Google "Toby's book does a great job of breaking down the complex subject matter of machine-learning algorithms into practical, easy-to-understand examples that can be directly applied to analysis of social interaction across the Web today. If I had this book two years ago, it would have saved precious time going down some fruitless paths." -- Tim Wolters, CTO, Collective Intellect

Book Information

Paperback: 362 pages

Publisher: O'Reilly Media; 1st edition (August 26, 2007)

Language: English
ISBN-10: 0596529325
ISBN-13: 978-0596529321
Product Dimensions: 7 x 0.9 x 9.2 inches
Shipping Weight: 1.6 pounds (View shipping rates and policies)
Average Customer Review: 4.4 out of 5 starsÂ Â See all reviewsÂ (120 customer reviews)
Best Sellers Rank: #56,095 in Books (See Top 100 in Books) #29 inÂ Books > Computers & Technology > Programming > Algorithms #35 inÂ Books > Computers & Technology > Computer Science > AI & Machine Learning > Intelligence & Semantics #332 inÂ Books > Computers & Technology > Programming > Languages & Tools

Customer Reviews

This book is probably best for those of you who have read the theory, but are not quite sure how to turn that theory into something useful. Or for those who simply hunger for a survey of how machine learning can be applied to the web, and need a non-mathematical introduction.My area of strength happens to be neural networks (my MS thesis topic was in the subject), so I will focus on that. In a few pages of the book, the author describes how the most popular of all neural networks, backpropagation, can be used to map a set of search terms to a URL. One might do this, for example, to try and find the page best matching the search terms. Instead of doing what nearly all other authors will do, prove the math behind the backprop training algorithm, he instead mentions what it does, and goes on to present python code that implements the stated goal.The upside of the approach is clear -- if you know the theory of neural networks, and are not sure how to apply it (or want to see an example of how it can be applied), then this book is great for that. His example of adaptively training a backprop net using only a subset of the nodes in the network was interesting, and I learned from it. Given all the reading I have done over the years on the subject, that was a bit of a surprise for me.However, don’t take this book as being the "end all, be all" for understanding neural networks and their applications. If you need that, you will want to augment this book with writings that cover some of the other network architectures (SOM, hopfield, etc) that are out there. The same goes for the other topics that it covers.

Have you ever wondered how some of those "collective intelligence" sites work? How can suggest books that you’ll like based on your browsing history? How a search engine can rank and filter results? Toby Segaran does a very good job in revealing and teaching those types of algorithms in

his book Programming Collective Intelligence: Building Smart Web 2.0 Applications. While I'm not ready to run out and build my own version of Facebook now, at least I can start to understand how sites like that are designed.Contents:Introduction to Collective Intelligence; Making Recommendations; Discovering Groups; Searching and Ranking; Optimization; Document Filtering; Modeling with Decision Trees; Building Price Models; Advanced Classification - Kernel Methods and SVMs; Finding Independent Features; Evolving Intelligence; Algorithm Summary; Third-Party Libraries; Mathematical Formulas; IndexIn each of the chapters, Segaran takes a type of capability, be it decision-making or filtering, and shows how a programming language can be used to build that feature. His examples are all in Python, so it helps if you are already familiar with that language if you want to actually work with the code. But even if you don't know Python, the examples are clear and detailed enough that you can follow along and get the gist of what's happening. I personally think that it would help immensely if you had a background in mathematics and statistics. You can use the code here without having a detailed understanding of math, but I'm sure much of this would be more deeply appreciated if you already know about such things as Tanimoto similarity scores, Euclidean distances, or Pearson coefficients.

& Data ... engineering, r programming, iOS development) Javascript Artificial Intelligence: Made Easy, w/ Essential Programming; Create your * Problem Solving * Algorithms! TODAY! w/ Machine Learning & Data ... engineering, r programming, iOS development) Artificial Intelligence: Made Easy w/ Ruby Programming; Learn to Create your * Problem Solving * Algorithms! TODAY! w/ Machine Learning & Data ... engineering, r programming, iOS development) High Performance ISAPI/NSAPI Web Programming: Your Complete Guide to Creating Fast, Powerful Web Server Programs AJAX and PHP: Building Responsive Web Applications Java(TM) Server and Servlets: Building Portable Web Applications e-RPG: Building AS/400 Web Applications with RPG React: Up & Running: Building Web Applications Application Reengineering: Building Web-Based Applications and Dealing with Legacies ASP.NET MVC 5 with Bootstrap and Knockout.js: Building Dynamic, Responsive Web Applications