

# Mahout

## IN ACTION

Sean Owen  
Robin Anil  
Ted Dunning  
Ellen Friedman



MEAP



**MEAP Edition  
Manning Early Access Program  
Mahout in Action version 7**

Copyright 2011 Manning Publications

For more information on this and other Manning titles go to  
[www.manning.com](http://www.manning.com)

# *Table of Contents*

1. Meet Apache Mahout

## ***Part 1 Recommendations***

2. Introducing recommenders

3. Representing data

4. Making recommendations

5. Taking recommenders to production

6. Distributing recommendation computations

## ***Part 2 Clustering***

7. Introduction to clustering

8. Representing data

9. Clustering algorithms in Mahout

10. Evaluating clustering quality

11. Taking clustering to production

12. Real-world applications of clustering

## ***Part 3 Classification***

- 13. Introduction to classification
- 14. Training a classifier
- 15. Evaluating and tuning a classifier
- 16. Deploying a classifier
- 17. Case study: Shop it To Me

## ***Appendices***

- A. JVM tuning
- B. Mahout math
- C. Resources

# 1

## Meet Apache Mahout

This chapter covers:

- What Apache Mahout is, and where it came from
- A glimpse of recommender engines, clustering, classification in the real world
- Setting up Mahout

As you may have guessed from the title, this book is about putting a particular tool, Apache Mahout, to effective use in real life. It has three defining qualities.

First, Mahout is an open source *machine learning* library from Apache. The algorithms it implements fall under the broad umbrella of “machine learning,” or “collective intelligence.” This can mean many things, but at the moment for Mahout it means primarily collaborative filtering / recommender engines, clustering, and classification.

It is also *scalable*. Mahout aims to be the machine learning tool of choice when the data to be processed is very large, perhaps far too large for a single machine. In its current incarnation, these scalable implementations are written in Java, and some portions are built upon Apache’s Hadoop distributed computation project.

Finally, it is a *Java library*. It does not provide a user interface, a pre-packaged server, or installer. It is a framework of tools intended to be used and adapted by developers.

To set the stage, this chapter will take a brief look at the sorts of machine learning that Mahout can help you perform on your data – recommender engines, clustering, classification – by looking at some familiar real-world instances.

In preparation for hands-on interaction with Mahout throughout the book, you will also step through some necessary setup and installation that will prepare you to work with the project.

### 1.1 Mahout’s Story

First, some background on Mahout itself is in order. You may be wondering how to say “Mahout” – as it is commonly Anglicized, it should rhyme with “trout.” It is a Hindi word that refers to an elephant driver, and to explain that one, here’s a little history. Mahout began life in 2008 as a subproject of Apache’s Lucene project, which provides the well-known open-source search engine of the same name. Lucene provides advanced implementations of search, text mining and information retrieval techniques. In the universe of Computer Science, these concepts are adjacent to machine learning techniques like clustering and, to an extent, classification. So, some of the work of the Lucene committers that fell more into these machine learning areas was spun off into its own subproject. Soon after, Mahout absorbed the “Taste” open-source collaborative filtering project.

Figure 1.1 shows some of Mahout’s lineage within the Apache Foundation. As of April 2010, Mahout has become a top-level Apache project in its own right, and got a brand-new elephant rider logo to boot.

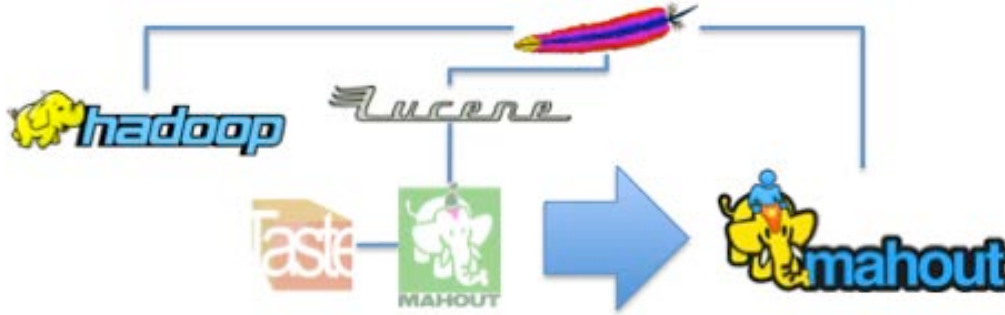


Figure 1.1 Apache Mahout and its related projects within the Apache Foundation.

Much of Mahout's work has been to not only implement these algorithms conventionally, in an efficient and scalable way, but also to convert some of these algorithms to work at scale on top of Hadoop. Hadoop's mascot is an elephant, which at last explains the project name!

Mahout incubates a number of techniques and algorithms, many still in development or in an experimental phase. At this early stage in the project's life, three core themes are evident: *collaborative filtering / recommender engines*, *clustering*, and *classification*. This is by no means all that exists within Mahout, but are the most prominent and mature themes at the time of writing. These therefore are the scope of this book.

Chances are that if you are reading this, you are already aware of the interesting potential of these three families of techniques. But just in case, read on.

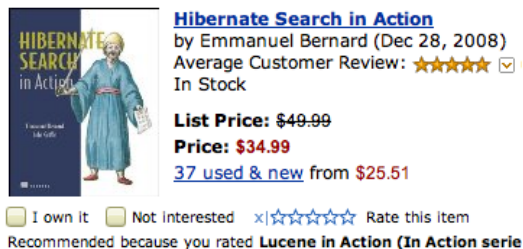
## 1.2 Mahout's Machine Learning Themes

While Mahout is, in theory, a project open to implementations of all kinds of machine learning techniques, it is in practice a project that focuses on three key areas of machine learning at the moment. These are recommender engines (collaborative filtering), clustering, and classification.

### 1.2.1 Recommender Engines

Recommender engines are the most immediately recognizable machine learning technique in use today. You will have seen services or sites that attempt to recommend books or movies or articles based on our past actions. They try to infer tastes and preferences and identify unknown items that are of interest:

- Amazon.com is perhaps the most famous commerce site to deploy recommendations. Based on purchases and site activity, Amazon recommends books and other items likely to be of interest. See Figure 1.2.
- Netflix similarly recommends DVDs that may be of interest, and famously offered a \$1,000,000 prize to researchers that could improve the quality of their recommendations.
- Dating sites like Libimseti (discussed later) can even recommend people to people.
- Social networking sites like Facebook use variants on recommender techniques to identify people most likely to be an as-yet-unconnected friend.



**Hibernate Search in Action**  
by Emmanuel Bernard (Dec 28, 2008)  
Average Customer Review: ★★★★★  
In Stock

**List Price: \$49.99**  
**Price: \$34.99**  
[37 used & new from \\$25.51](#)

I own it  Not interested  ★★★★★ Rate this item

Recommended because you rated **Lucene in Action (In Action serie)**

Figure 1.2 A recommendation from Amazon. Based on past purchase history and other activity of customers like the user, Amazon considers this to be something the user is interested in. It can even tell the user something similar that he or she has bought or liked that in part caused the recommendation.

As Amazon and others have demonstrated, recommenders can have concrete commercial value too, by enabling smart cross-selling opportunities. One firm reports that recommending products to users can drive an 8-12% increase in sales<sup>1</sup>.

### 1.2.2 Clustering

Clustering turns up in less apparent but equally well-known contexts. As its name implies, clustering techniques attempt to group a large number of things together into clusters that share some similarity. It is a way to discover hierarchy and order in a large or hard-to-understand data set, and in that way reveal interesting patterns or make the data set easier to comprehend.

- Google News groups news articles according to their topic using clustering techniques in order to present news grouped by logical story, rather than a raw listing of all articles. Figure 1.3 below illustrates this.
- Search engines like Clusty group search results for similar reasons
- Consumers may be grouped into segments (clusters) using clustering techniques based on attributes like income, location, and buying habits.

#### **Obama to Name 'Smart Grid' Projects**

Wall Street Journal - [Rebecca Smith](#) - 1 hour ago

The Obama administration is expected Tuesday to name 100 utility projects that will share \$3.4 billion in federal stimulus funding to speed deployment of advanced technology designed to cut energy use and make the electric-power grid ...

[Cobb firm wins "smart-grid" grant](#) Atlanta Journal Constitution  
[Obama putting \\$3.4B toward a 'smart' power grid](#) The Associate  
[Baltimore Sun](#) - [Bloomberg](#) - [New York Times](#) - [Reuters](#)  
[all 594 news articles »](#) [Email this story](#)

Figure 1.3. A sample news grouping from Google News. A detailed snippet from one representative story is displayed, and links to a few other similar stories within the cluster for this topic are shown. Links to all the rest of the stories that clustered together in this topic are available too.

Clustering helps discover structure, and even hierarchy, among a large collection of things which may be otherwise difficult to make sense of. Enterprises might use this to discover hidden groupings among users, or organize a large collection of documents sensibly, or discover common usage patterns for a site based on logs.

### 1.2.3 Classification

Classification techniques decide how much a thing is or isn't part of some type or category, or, does or doesn't have some attribute. Classification is likewise ubiquitous, though even more behind-the-scenes. Often these systems "learn" by reviewing many instances of items of the categories in question in order to deduce classification rules. This general idea finds many applications:

<sup>1</sup> <http://www.practicalearcommerce.com/articles/1942-10-Questions-on-Product-Recommendations>

- Yahoo! Mail decides whether incoming messages are spam, or not, based on prior emails and spam reports from users, as well as characteristics of the e-mail itself. A few messages classified as spam are shown in Figure 1.4.
- Picasa (<http://picasa.google.com/>) and other photo management applications can decide when a region of an image contains a human face.
- Optical character recognition software classifies small regions of scanned text into individual characters by classifying the small areas as individual characters.
- Apple's Genius feature in iTunes reportedly uses classification to classify songs into potential playlists for users



 Spam (49)	Empty	<input type="checkbox"/>	Hevnerco	DishView	Wed 10/28, 12:34 PM
 Trash	Empty	<input type="checkbox"/>	Customer Service	FINAL NOTIFICATION:..Please r...	Wed 10/28, 4:53 AM
Contacts	Add	<input type="checkbox"/>	MmddDdhb	From: MmddDdhb Read The File.	Wed 10/28, 12:58 AM

Figure 1.4 Spam messages as detected by Yahoo! Mail. Based on reports of email spam from users, plus other analysis, the system has learned certain attributes that usually identify spam. For example, messages mentioning “viagra” are frequently spam – as are those with clever misspellings like “v1agra”. The presence of such terms are an example of an attribute that a spam classifier can learn.

Classification helps decide whether a new input or thing matches a previously observed pattern or not, and is often used to classify behavior or patterns as unusual. It could be used to detect suspicious network activity or fraud. It might be used to figure out when a user's message indicates frustration or satisfaction.

Each of these techniques works best when provided with a large amount of good input data. In some cases, these techniques must work not only on large amounts of input, but must produce results quickly. These factors quickly make scalability a major issue. And, as mentioned before, one of Mahout's key reasons for being is to produce implementations of these techniques that do scale up to huge input.

### 1.3 Tackling large scale with Mahout and Hadoop

How real is the problem of scale in machine learning algorithms? Let's consider a few examples of the size of problems where you might deploy Mahout.

Consider that Picasa may have hosted over half a billion photos even three years ago, according to some crude estimates<sup>2</sup>. This implies millions of new photos per day that must be analyzed. The analysis of one photo by itself is not a large problem, though it is repeated millions of times. But, the learning phase can require information from each of the billions of photos simultaneously -- a computation on a scale that is not feasible for a single machine.

According to a similar analysis, Google News sees about a 3.5 million new news articles *per day*. Although this by itself is not a large amount, consider that these articles must be clustered, along with other recent articles, in *minutes* in order to become available in a timely manner.

The subset of rating data that Netflix published for the Netflix Prize contained 100 million ratings<sup>3</sup>. Since this was just the data released for contest purposes, presumably, the total amount of data that Netflix actually has and must process to create recommendations is many times larger!

These techniques are necessarily deployed in contexts where the amount of input is large – so large, that it is not feasible to process it all on one computer, even a powerful one. Without an implementation such as Mahout, these would be impossible tasks. This is why Mahout makes scalability a top priority, and, why this book will focus, in a way that others don't, on dealing with large data sets effectively.

Sophisticated machine learning techniques, applied at scale, were until recently only something that large, advanced technology companies could consider using. But today computing power is cheaper than ever and more accessible via open-source frameworks like Apache's Hadoop. Mahout attempts to

<sup>2</sup> <http://blogoscoped.com/archive/2007-03-12-n67.html>

<sup>3</sup> <http://archive.ics.uci.edu/ml/machine-learning-databases/netflix/>

complete the puzzle by providing quality, open-source implementations capable of solving problems at this scale, with Hadoop, and putting this into the hands of all technology organizations.

Some of Mahout makes use of Hadoop, which includes an open-source, Java-based implementation of the MapReduce (<http://labs.google.com/papers/mapreduce.html>) distributed computing framework popularized and used internally at Google. MapReduce is a programming paradigm that at first sounds odd, or too simple to be powerful. The MapReduce paradigm applies to problems where the input is a set of key-value pairs. A “map” function turns these key-value pairs into other intermediate key-value pairs. A “reduce” function merges in some way all values for each intermediate key, to produce output. Actually, many problems can be framed as a MapReduce problem, or a series of them. And, the paradigm lends itself quite well to parallelization: all of the processing is independent, and so can be split across many machines. Rather than reproduce a full explanation of MapReduce here, we refer you to tutorials such as the one provided by Hadoop ([http://hadoop.apache.org/common/docs/current/mapred\\_tutorial.html](http://hadoop.apache.org/common/docs/current/mapred_tutorial.html)).

Hadoop implements the MapReduce paradigm, which is no small feat, even given how simple MapReduce sounds. It manages storage of the input, intermediate key-value pairs, and output; this data could potentially be massive, and, must be available to many worker machines, not just stored locally on one. It manages partitioning and data transfer between worker machines. It handles detection of and recovery from individual machine failure. Understanding how much work goes on behind the scenes will help prepare you for how relatively complex using Hadoop can seem. It’s not just a library you add to your project. It’s several components, each with libraries and (several) standalone server processes, which might be run on several machines. Operating processes based on Hadoop is not simple, but, investing in a scalable, distributed implementation can pay dividends later: because your data may grow exponentially to great sizes before you know it, this sort of scalable implementation is a way to future-proof your application.

Later, this book will try to cut through some of that complexity to get you running on Hadoop fast, at which point you can explore the finer points and details of operating full clusters, and tuning the framework. Because this is complex framework that needs a great deal of computing power is becoming so popular, it’s not surprising that cloud computing providers are beginning to offer Hadoop-related services. For example Amazon offers Elastic MapReduce (<http://aws.amazon.com/elasticmapreduce/>), a service which manages a Hadoop cluster, provides the computing power, and puts a friendlier interface on the otherwise complex task of operating and monitoring a large-scale job with Hadoop.

## 1.4 Setting up Mahout

You will need to assemble some tools before you can “play along at home” as we present some code in the coming chapters. We assume you are comfortable with Java development already.

Mahout and its associated frameworks are Java-based and therefore platform-independent, so you should be able to use it with any platform that can run a modern JVM. At times, we will need to give examples or instructions that will vary from platform to platform. In particular, command-line commands are somewhat different in a Windows shell than in a FreeBSD `tcsh` shell. We will use commands and syntax that work with `bash`, a shell found on most Unix-like platforms. This is the default on most Linux distributions, Mac OS X, many Unix variants, and Cygwin (a popular Unix-like environment for Windows). Windows users who wish to use the Windows shell are the most likely to be inconvenienced by this. Still, it should be simple to interpret and translate the listings given in this book to work for you.

### 1.4.1 Java and IDE

Java is likely already installed on your personal computer if you have done any Java development so far. Note that Mahout requires Java 6. If in doubt, open a terminal and type `java -version`. If the reported version does not begin with “1.6”, you need to also install Java 6.

Windows and Linux users can find a Java 6 JVM from Sun at <http://java.sun.com>. Apple provides a Java 6 JVM for Mac OS X 10.5 and 10.6. If it does not appear that Java 6 is being used, open “Java Preferences” under `/Applications/Utilities`. This will allow you to select Java 6 as the default.

Most people will find it quite a bit easier to edit, compile and run the many examples to come with the help an IDE; this is *strongly* recommended. Eclipse (<http://www.eclipse.org>) is the most popular, free

Java IDE. Installing and configuring Eclipse is beyond the scope of this book, but you should spend some time becoming familiar with it before proceeding. NetBeans (<http://netbeans.org/>) is also a popular, free IDE. IntelliJ IDEA (<http://www.jetbrains.com/idea/index.html>) is another powerful and popular IDE, with a free “community” version now available.

For example, IDEA can create a new project from an existing Maven model; by specifying the root directory of the Mahout source code upon creating a project, it will automatically configure and present the entire project in an organized manner. It’s then possible, for example, drop the source code found throughout this book under the `core/src/...` source root, and run it from within IDE with one click -- the details of dependencies and compilation are managed automatically. This should prove far easier than attempting to compile and run manually.

### 1.4.2 Installing Maven

As with many Apache projects, Mahout’s build and release system is built around Maven (<http://maven.apache.org>). Maven is a command-line tool that manages compiling code, packaging release, generating documentation, and publishing formal releases. Although it has some superficial resemblances to the also-popular Ant build tool, it is not the same. Ant is a flexible, lower-level scripting language, and Maven is a higher-level tool more purpose-built for release management.

Because Mahout uses Maven, you should install Maven yourself. Mac OS X users will be pleased to find that Maven should already be installed. If not, install Apple’s Developer Tools. Type `mvn --version` on the command line. If you successfully see a version number, and the version is at least 2.2, you are ready to go. If not, you should install a local copy of Maven.

Users of Linux distributions with a decent package management system may be able to use it to quickly obtain a recent version of Maven. Otherwise, standard procedure would be to download a binary distribution, unpack it to a common location such as `/usr/local/maven`, then edit bash’s configuration file, `~/.bashrc`, to include a line like `export PATH=/usr/local/maven/bin:$PATH`. This will ensure that the `mvn` command is always available.

If you are using an IDE like Eclipse or IntelliJ, it already includes Maven integration. Refer to its documentation to learn how to enable the Maven integration. This will make working with Mahout in an IDE much simpler, as the IDE can use the project’s Maven configuration file (`pom.xml`) to instantly configure and import the project.

### 1.4.3 Installing Mahout

Mahout is still in development. This book was written to work with the 0.4 release of Mahout. This release and others may be downloaded by following instructions at <https://cwiki.apache.org/confluence/display/MAHOUT/Downloads>; the archive of source code may be unpacked anywhere that is convenient on your computer.

Because Mahout is changing frequently, and bug fixes and improvements are added regularly, it may be useful in practice to use a later release (or even the latest, unreleased code from Subversion. See <https://cwiki.apache.org/confluence/display/MAHOUT/Version+Control>). Future point releases should be backwards-compatible with the examples in this book.

Once you have obtained the source, either from Subversion or from a release archive, create a new project for Mahout in your IDE. This is IDE-specific; refer to its documentation for particulars of how this is accomplished. It will be easiest to use your IDE’s Maven integration to simply import the Maven project from the `pom.xml` file in the root of the project source.

Once configured, you can easily create a new source directory within this project to hold sample code that will be introduced in upcoming chapters. With the project properly configured, you should be able to compile and run the code transparently with no further effort.

### 1.4.4 Installing Hadoop

For some activities later in this book, you will need your own local installation of Hadoop. You do not need a cluster of computers to run Hadoop. Setting up Hadoop is not difficult, but not trivial. Rather than repeat the procedures, we direct you to obtain a copy of Hadoop version 0.20.2 from the Hadoop web site at

<http://hadoop.apache.org/common/releases.html>, and then set up Hadoop for “pseudo-distributed” operation by following the quick start documentation currently found at <http://hadoop.apache.org/common/docs/current/quickstart.html>.

## **1.5 Summary**

Mahout is a young, open-source, scalable machine learning library from Apache, and this book is a practical guide to using Mahout to solve real problems with machine learning techniques. In particular, you will soon explore recommender engines, clustering, and classification. If you’re a researcher familiar with machine learning theory and looking for a practical how-to guide, or a developer looking to quickly learn best practices from practitioners, this book is for you.

These techniques are no longer merely theory: we’ve noted already well-known examples of recommender engines, clustering, and classification deployed in the real world: e-commerce, e-mail, videos, photos and more involve large-scale machine learning. These techniques have been deployed to solve real problems and even generate value for enterprises -- and are now accessible via Mahout.

And, we’ve noted the vast amount of data sometimes employed with these techniques – scalability is a uniquely persistent concern in this area. We took a first look at MapReduce and Hadoop and how they power some of the scalability that Mahout provides.

Because this will be a hands-on, practical book, we’ve set up to begin working with Mahout right away. At this point, you should have assembled the tools you will need to work with Mahout and be ready for action. Because this book intends to be practical, let that wrap up the opening remarks now and get on to some real code with Mahout. Read on!