

# **Implementing Scalable and High Performance Machine Learning Algorithms using Apache Mahout**

*Ankita*

*Research Scholar*

*Department of Computer Science*

*OPJS University, Rajasthan, India*

*Dr. Om Parkash*

*Associate Professor*

*Department of Computer Science*

*OPJS University, Rajasthan, India*

## **Abstract**

Machine learning refers to the intelligent and dynamic response by the software or embedded hardware programs depending upon the input data. Machine learning is the specialized domain that operates in association with the artificial intelligence to have strong predictions and analysis. Using this approach, there is no need to explicitly program the computers for specific applications rather the computing modules evaluates the dataset with its inherent behavior so that real time fuzzy based analysis can be done. The programs developed with machine learning paradigms focuses on the dynamic input and dataset so that the custom and related output can be presented to the end user.

*Keywords : Apache Mahout, High Performance Computing Machine Learning*

### **Introduction**

A number of application domains exist where machine learning approaches are widely used including fingerprint analysis, multidimensional biometric evaluation, image forensic, pattern recognition, criminal investigation, bioinformatics, Biomedical informatics, Computer vision, Customer relationship management, Data mining, Email filtering, Natural language processing, Automatic summarization, Automatic taxonomy construction, Robotics, Dialog system, Grammar checker, Language recognition, Handwriting recognition, Optical character recognition, Speech recognition, Machine translation, Question answering, Speech synthesis, Text simplification, Pattern recognition, Facial recognition system, Handwriting recognition, Image recognition, Search engine analytics, Recommendation system and many others [1].

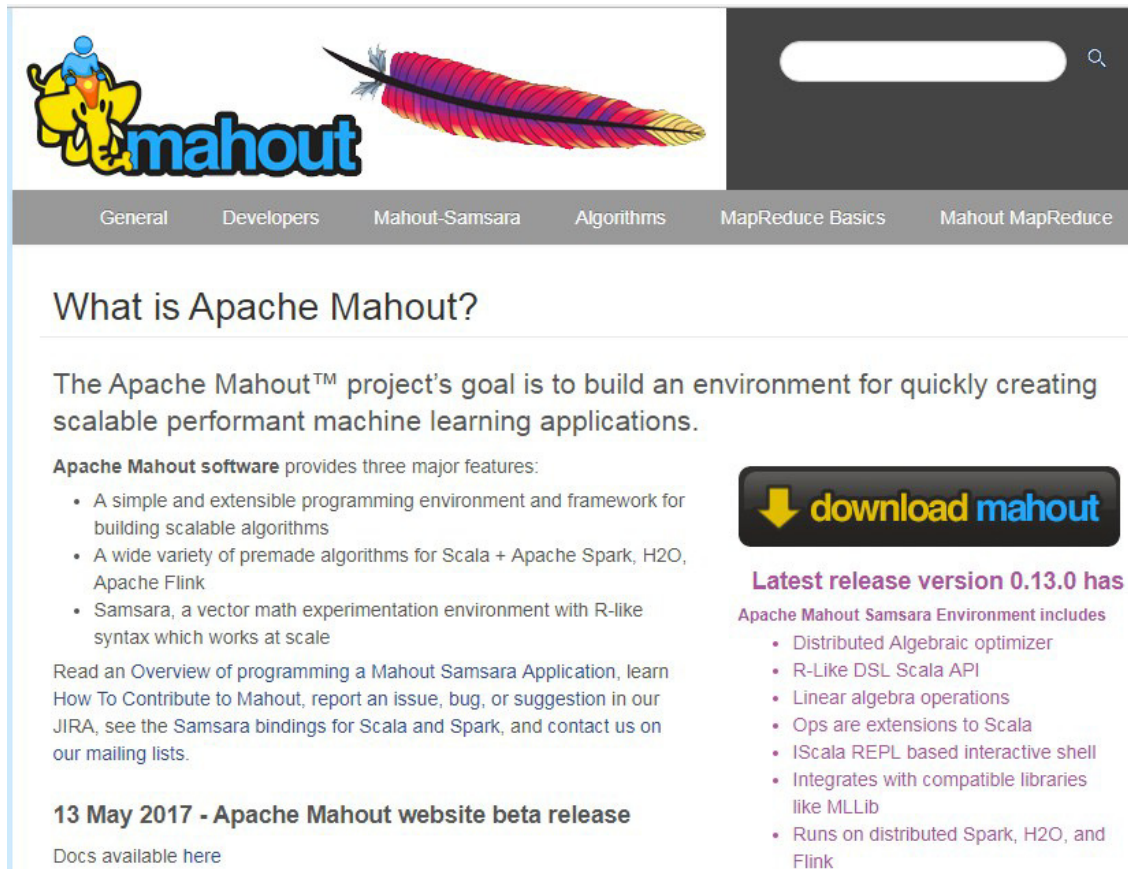
A number of approaches are implemented to machine learning but in traditional integrations the Supervised and Unsupervised Learning is widely used. In supervised learning, the program is trained with a specific type of dataset with the target value. After learning and deep evaluation of the input data and corresponding target, it starts giving prediction. The common examples of supervised learning algorithms include artificial neural networks, support vector machines and the classifiers. In case of unsupervised learning, the target is not assigned with the input data. In this approach, the dynamic evaluation of data is done with the high performance algorithms including k-means, self organizing maps (SOM) [2] and clustering techniques. Other prominent approaches and algorithms associated with Machine Learning includes Dimensionality reduction, Decision tree algorithm, Ensemble learning, Regularization algorithm, Supervised learning, Artificial neural network, Deep learning, Instance-based algorithm, Regression analysis, Classifiers, Bayesian statistics, Linear classifier, Unsupervised learning, Artificial neural network, Association rule learning, Hierarchical clustering, deep cluster evaluation, Anomaly detection, Semi-supervised learning, Reinforcement learning and many others [3].

**Free and Open Source Tools for Machine Learning**

- Apache Mahout
- Scikit-Learn
- OpenAI
- TensorFlow
- Char-RNN
- PaddlePaddle
- CNTX
- Apache Singa
- DeepLearning4J
- H2O
- GNU Octave
- R
- Orange
- WEKA
- Torch
- Yooreeka
- Shogun
- Massive Online Analysis (MOA)
- Mallet
- ELKI

**Apache Mahout: The Scalable High Performance Machine Learning Framework**

URL: [mahout.apache.org](http://mahout.apache.org)



The screenshot shows the Apache Mahout website. At the top left is the Mahout logo, which features a yellow elephant with a blue person riding on its back, and the word "mahout" in blue lowercase letters. To the right of the logo is a large, colorful feather. Below the logo and feather is a navigation menu with the following items: General, Developers, Mahout-Samsara, Algorithms, MapReduce Basics, and Mahout MapReduce. To the right of the navigation menu is a search bar. Below the navigation menu is the main content area. The main heading is "What is Apache Mahout?". Below this heading is a paragraph: "The Apache Mahout™ project's goal is to build an environment for quickly creating scalable performant machine learning applications." Below this paragraph is a section titled "Apache Mahout software provides three major features:" followed by a bulleted list: "A simple and extensible programming environment and framework for building scalable algorithms", "A wide variety of premade algorithms for Scala + Apache Spark, H2O, Apache Flink", and "Samsara, a vector math experimentation environment with R-like syntax which works at scale". Below the bulleted list is a paragraph: "Read an Overview of programming a Mahout Samsara Application, learn How To Contribute to Mahout, report an issue, bug, or suggestion in our JIRA, see the Samsara bindings for Scala and Spark, and contact us on our mailing lists." Below this paragraph is a section titled "13 May 2017 - Apache Mahout website beta release" followed by a link "Docs available here". To the right of the main content area is a dark blue button with a yellow arrow pointing down and the text "download mahout". Below the button is a section titled "Latest release version 0.13.0 has" followed by a paragraph: "Apache Mahout Samsara Environment includes" followed by a bulleted list: "Distributed Algebraic optimizer", "R-Like DSL Scala API", "Linear algebra operations", "Ops are extensions to Scala", "IScala REPL based interactive shell", "Integrates with compatible libraries like MLlib", and "Runs on distributed Spark, H2O, and Flink".

**Figure 1: Official Portal of Apache Mahout**

Apache Mahout [4] is the powerful and high performance machine learning framework for the implementation of machine learning algorithms. Apache Mahout is traditionally used for the integration of supervised machine learning algorithms with the target value assigned to each input data set. Apache Mahout can be used for assorted research based applications including Social Media Extraction and Sentiment Mining, User Belief Analytics, YouTube Analytics and many related real time applications.

In Apache Mahout, a Mahout refers to the object which drives or operates the elephant. The mahout act as the master of elephant in association with Apache Hadoop and it is presented in the logo of elephant. Apache Mahout runs with the base installation of Apache Hadoop and then the machine learning algorithms are implemented with the features to develop and deploy the scalable machine learning algorithms. The prime approaches like recommender engines, classification problems and clustering can be effectively solved using mahout.

Corporate Users of Mahout includes the following

- Adobe
- Facebook
- LinkedIn
- FourSquare
- Twitter
- Yahoo

### **Installation of Apache Mahout**

To start with the Mahout installation, first of all Apache Hadoop is required to be setup on the Linux Distribution. To get ready with Hadoop, the installation is required to be updated as follows in the Ubuntu Linux.

```
$ sudo apt-get update
```

```
$ sudo addgroup hadoop
```

```
$ sudo adduser --ingroup hadoop hadoopuser1
```

```
$ sudo adduser hadoopuser1 sudo
```

```
$ sudo apt-get install ssh
```

```
$ su hadoopuser1
```

```
$ ssh-keygen -t rsa
```

```
$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

```
$ chmod 0600 ~/.ssh/authorized_keys
```

```
$ ssh localhost
```

### **Installing the Latest Version of Hadoop**

```
$ wget http://www-us.apache.org/dist/hadoop/common/hadoop-HadoopVersion/hadoop-HadoopVersion.tar.gz
```

```
$ tar xvzf hadoop-HadoopVersion.tar.gz
```

```
$ sudo mkdir -p /usr/local/hadoop
```

```
$ cd hadoop-HadoopVersion/
```

```
$ sudo mv * /usr/local/hadoop
```

```
$ sudo chown -R hadoopuser1:hadoop /usr/local/hadoop
```

The following files are required to be updated next

- `~/.bashrc`
- `core-site.xml`
- `hadoop-env.sh`
- `hdfs-site.xml`
- `mapred-site.xml`
- `yarn-site.xml`

```
$ hadoop namenode -format
```

```
$ cd /usr/local/hadoop/sbin
```

```
$ start-all.sh
```

### **Web Interfaces of Hadoop**

```
MapReduce: http://localhost:8042/
```

*NameNode daemon: http://localhost:50070/*

*Resource Manager: http://localhost:8088/*

*SecondaryNameNode:: http://localhost:50090/status.html*

The default port to access Hadoop is 50070 and using <http://localhost:50070/> on Web Browser

After installation of Hadoop, the setup of Mahout is required as follows.

```
$ wget http://mirror.nexcess.net/apache/mahout/0.9/mahout-Distribution.tar.gz
```

```
$ tar zxvf mahout-Distribution.tar.gz
```

### **Implementation of Recommender Engine Algorithm**


Now days, we shop on the online shopping platforms like Amazon, E-Bay, SnapDeal, FlipKart and many others. We generally see that most of these online shopping platforms give us suggestions or recommendations about the products which we like or earlier purchased. This type of implementation or suggestive modeling is known as recommender engine or recommendation system. Even in YouTube, we see the number of suggestions regarding related videos which we view. Such online platforms integrate the approaches of recommendation engines by which the related best fit or most viewed items are presented to the user as recommendations.

Apache Mahout provides the platform to program and implement the recommender systems. For example, the Twitter HashTag Popularity can be evaluated and ranking can be done based on the visitor count or popularity or simply hits by the users. In YouTube, the number of viewers is the key value which determines the actual popularity of that particular video.

Using Apache Mahout, such algorithms can be implemented which are covered under high performance real time machine learning.

For example, a data table which presents the popularity of products after online shopping by the users is recorded by the companies so that the overall analysis of popularity of products can be done. The rating from 0-5 is logged from the users so that the overall prominence of the product can be evaluated. This dataset can be evaluated using Apache Mahout in Eclipse IDE.


For integration of Java Code with Apache Mahout Libraries on Eclipse IDE, there are specific JAR files which are required to be added from Simple Logging Facade for Java (SLF4J).



SLF4J Project	<h2>Simple Logging Facade for Java (SLF4J)</h2> <p>The Simple Logging Facade for Java (SLF4J) serves as a simple facade or abstraction for various logging frameworks (e.g. java.util.logging, logback, log4j) allowing the end user to plug in the desired logging framework at <i>deployment</i> time.</p> <p>Before you start using SLF4J, we highly recommend that you read the two-page <a href="#">SLF4J user manual</a>.</p> <p>Note that SLF4J-enabling your library implies the addition of only a single mandatory dependency, namely <i>slf4j-api.jar</i>. If no binding is found on the class path, then SLF4J will default to a no-operation implementation.</p> <p>In case you wish to migrate your Java source files to SLF4J, consider our <a href="#">migrator tool</a> which can migrate your project to use the SLF4J API in just a few minutes.</p> <p>In case an externally-maintained component you depend on uses a logging API other than SLF4J, such as commons logging, log4j or java.util.logging, have a look at SLF4J's binary-support for <a href="#">legacy APIs</a>.</p> <hr/> <p>Copyright © 2004-2017 QOS.ch          We are actively looking for volunteers to proofread the documentation. Please send your corrections or suggestions for improvement to "corrections@qos.ch". See also the <a href="#">instructions for contributors</a>.</p>
Introduction	
Download	
Documentation	
License	
News	
Support	
Mailing Lists	
Bug Reporting	
Source Repository	
Support offerings	
Native implementations	
Logback	
Wrapped implementations	
JDK14	
Log4j	
Simple	

**Figure 2: Simple Logging Facade for Java**





---

SLF4J Project
Introduction
Download
Documentation
License
News
Support
Mailing Lists
Bug Reporting
Source Repository
Support offerings
Native implementations
Logback
Wrapped implementations
JDK14
Log4j
Simple

### Latest STABLE version

Download version 1.7.25 including *full source code*, class files and documentation in ZIP or TAR.GZ format:

- [slf4j-1.7.25.tar.gz](#)
- [slf4j-1.7.25.zip](#)

### Java 9 Modularized EXPERIMENTAL version

Download version 1.8.0-alpha2 including *full source code*, class files and documentation in ZIP or TAR.GZ format:

- [slf4j-1.8.0-alpha2.tar.gz](#)
- [slf4j-1.8.0-alpha2.zip](#)

#### Previous versions

Previous versions of SLF4J can be downloaded from the [main repository](#).

**Figure 3: Stable JAR Files from SLF54J Portal**

Following is the Java Code Module with the methods which can be executed using Eclipse IDE with the JAR files of Mahout to implement Recommender Algorithm

```
DataModel dm = new FileDataModel(new File("inputdata"));
UserSimilarity us = new PearsonCorrelationSimilarity(dm);
UserNeighborhood un = new ThresholdUserNeighborhood(ThresholdValue), us, dm);
UserBasedRecommender r=new GenericUserBasedRecommender(dm, un, us);
List<RecommendedItem> rs=recommender.recommend(UserID, Recommendations);
for (RecommendedItem rc : rs) {
System.out.println(rc);
```

### **Conclusion**

The research problems can be solved effectively using Apache Mahout with the customized algorithms in multiple applications including Malware Predictive Analytics, User Sentiment Mining, Rainfall Predictions, Network Forensic and Network Routing with deep analytics. Now days, the integration of deep learning approaches can be embedded in the existing algorithms so that higher degree of accuracy and optimization in the results can be achieved.

### **References**

- [1] Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.
- [2] Van Gassen, S., Callebaut, B., Van Helden, M. J., Lambrecht, B. N., Demeester, P., Dhaene, T., & Saeys, Y. (2015). FlowSOM: Using self-organizing maps for visualization and interpretation of cytometry data. *Cytometry Part A*, 87(7), 636-645.
- [3] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Kudlur, M. (2016, November). TensorFlow: A System for Large-Scale Machine Learning. In *OSDI* (Vol. 16, pp. 265-283).
- [4] Gupta, A. (2015). Learning Apache Mahout Classification. Packt Publishing Ltd.