ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 2, Issue 2, pp: (336-342), Month: April-June 2014, Available at: www.researchpublish.com

Mini-Batch K-Means Clustering Using Map-Reduce in Hadoop

Mr. Krishna Yadav
Computer Science and Engineering Department
Parul Institute of Engineering and Technology, Limda, Vadodara, India

Mr. Jwalant Baria,
Assistant professor, Computer Science and Engineering Department
Parul Institute of Engineering and Technology, Limda, Vadodara, India

Abstract: **The main objective is to describe an approach for data clustering by using Mini Batch K-Means algorithm. The implementation describes here optimizes the K-Means by using one-pass over the input data and produces as many centroids as it determines is optimal. Avoiding multiple passes over the input data can have major impacts on running time because just reading large data set can increase the cost in large-scale computations. Mini Batch K-Means algorithm is implemented by using Hadoop framework. Mini Batch K-Means is implemented using Map-Reduce programming paradigms and clusters of machine is created by using VMware virtual machine. Experimental results are compared between existing system K-Means and proposed system Mini Batch K-Means by using datasets like reuters21578 and SC time series dataset. Mini Batch K-Means clustering algorithm can improve parameters like accuracy at good extent as it shows compact and well-separated clusters and computation time can also decrease as compared to existing algorithm. Performance can also improve by using more number of machines in Hadoop cluster.**

Keywords: **Mini Batch K-Means; hadoop; K-Means; Map-Reduce; Clustering.**

I. INTRODUCTION

Data clustering is the partitioning of a data set or sets of data into similar subsets. During the process of data clustering a method is often required to determine how similar one object or groups of objects is to another. This method is usually encompassed by some kind of distance measure. Data clustering is a common technique used in data analysis and is used in many fields including statistics, data mining and image analysis. There are many types of clustering algorithms. Clustering algorithms can also be partitioned meaning they determine all clusters at once [4]. Data clustering can be computationally expensive in terms of time and space complexity. In addition further expense may be incurred by the need to repeat data clustering. Hence, parallelizing and distributing expensive data clustering tasks becomes attractive in terms of speed-up of computation and the increased amount of memory available in a computing cluster. Map-Reduce are a software framework for solving certain kinds of distributable problems using a computing cluster. The Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. Hadoop Map-Reduce: A YARN-based system for parallel processing of large data sets [9].

K-means clustering is a method of vector quantization originally from signal processing that is popular for cluster analysis in data mining. K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. Now K-means work well when it routine with Map-Reduce through distributed programming and implement on Hadoop framework. Many further modifications are done in K-means to make it better and more efficient.

II. PROBLEM STATEMENT

No matter how good anything can be there is always scope for improvement. Primary area of improvement in any algorithm is its accuracy, speed and efficiency. K-means is good for clustering, but it having problems like low computation speed, sensitivity to outliers and inefficiency for large data sets and instability.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 2, Issue 2, pp: (336-342), Month: April-June 2014, Available at: www.researchpublish.com

Here, group n d-dimensional data-points into k-disjoint sets.

$$\sum_{C1}^{Ck} \left(\sum_{xij \in Xi} dist(Xij, Ci) \right)$$

Xi is the i^{th} cluster
Ci is the centroid of the i^{th} cluster
Xij is the j^{th} point from the i^{th} cluster
Dist(x, y) = $||x - y||_2$
The problem is that for each iteration, need to compute the distance between each data-points to each of the centroids until convergence point obtained for centroid and form cluster.

A. Motivation

Parallelizing and distributing expensive data clustering tasks becomes attractive in terms of speed-up of computation and the increased amount of memory available in a computing cluster. Programming distributed memory systems Message Passing Interface (MPI) is a widely used standard [5]. A disadvantage of MPI is that the programmer must have a sophisticated knowledge of parallel computing concepts such as deadlocks and synchronization. RDBMS is also good for distributed data-storage and process, but it fails to handle data of PB or TB. Map-Reduce is a software framework for solving certain kinds of distributable problems using a computing cluster [9]. There are many simple and standard data clustering Algorithm which can be used with Map-Reduce.

B. Objective

- To study Hadoop framework
- To study K-Means using Map-Reduce over distributed Environment in Mahout.
- To study Map-Reduce Programming and getting results from distributed data over network.
- Improve the K-Means clustering algorithm for speedy computation using various datasets.
- To evaluate the performance and accuracy of Mini Batch K-Means by using confusion matrix.

III. MINI BATCH K-MEANS USING MAP-REDUCE

A. Overview

Clustering is one of the most widely used techniques for exploratory data analysis and study. Across all disciplines, from general sciences over biology to computer science, people always try to get a first intuition about their data by finding meaningful groups among the data objects. K-Means is one of the most famous clustering algorithms for all time. K-Means is best algorithm for text clustering and document clustering. Its simplicity and speed allow it to run on large data-sets very efficiently. No matter how good any algorithm can be, but there is always a scope for improvement. However, it also has various drawbacks. First, this algorithm is slow at execution. Second, it has instability, low accuracy and sensibility towards outlier.

Proposed system assumed that when K-Means algorithm is modifies and optimizes the K-Means by using one-pass over the input data and produces as many centroids as it determines is optimal. Avoiding multiple passes over the input data can have major impacts on running time because just reading large data set can increase the cost in large-scale computations. The basic idea behind this algorithm is that incoming points are assigned to a nearby cluster or used as basis of new cluster. Decision is made according to how distance to nearby cluster compares to adaptive scale parameter, then the limitations of existing system can be overcome. Distributed computing framework Map-Reduce is there to solve the inefficiency problem in clustering on large data sets and improve its time-complexity.

The algorithm iterates between two major steps. In the first step, Streaming step is taken which is a randomized algorithm which makes single pass through the data and helps in producing as many centroids as it determines is optimal. Suppose, if the size of data stream is n and expected number of clusters is k, then the streaming step will produce roughly k*log (n)

clusters. Now the obtained clusters are passed to second step which is BallKMeans. BallKMeans will further reduce the number of clusters down to k, the expected number of clusters. Here clusters are formed by assigning each data points to the nearest centroids. First step is processed at Mapfunction, runs by Mapper and second step is processed by Reduce function, runs by Reducer.

B. Mini Batch K-Means Execution

The Mini Batch K-Means framework with Map-Reduce details are as follows:

1. Initial cutoff is calculated using a sample of points from the throughput of the input sequence.
2. Then the input data is split and further each split is passed to a mapper that consists of an instance of the streaming k-means algorithm.
3. The result will be a set of centroids C that is much larger than the final desired clusters K but, it should be small to fit into the memory.
4. Now, the results of clustering each split are passed to single reducer uses the streaming Ball k-means function to combine the clusters as needed.

The initial set of data samples is stored in the input directory of HDFS prior to Map routine call and they form the 'key' field in the <key,value> pair. Based on distance-cutoff parameters required to measure the distance between the point and the centroids, the point is either merged to any of the clusters or become a new centroid to form new cluster. The points which are forming the clusters with any of centroids become the value in <key, value> pair.

Now, the results of clustering each split from mapper are all passed to a single reducer. Here, reducer will work on BallKMeans instance. Clusters are formed by assigning each of data points to the nearest centroids. The center of mass of the trimmed clusters becomes the new cluster.

C. Experimental Parameters

1. Accuracy

Typical the main objective functions in clustering formalize is of attaining high intra-cluster similarity (means documents within a cluster are similar and compact) and lower inter-cluster similarity (means documents from different clusters are dissimilar). This is an internal criterion for the quality of a clustering. To compute purity, each cluster is assigned to the class which is most frequent in the cluster and closed to it, and then the accuracy of this assignment is measured by counting the number of correctly assigned documents and dividing by total number of assignment N.

2. Performance

Here, Map-Reduce programming paradigm is use for calculating the cluster. Map-Reduce are used for distributed programming over Hadoop cluster. Performance of the system can be increased by increasing number of machine, where actual process of clustering is run. Machines of various configurations can work together in the cluster for processing the data. Hence, platform is not an issue in Hadoop cluster.

3. Time

In traditional clustering algorithm most of the time is spend on calculating the distance between each of data points to each of centroids at each iteration and then closet points is assigned to nearest centroids. It is an NP-hard problem for clustering algorithms when they used to cluster high dimensional documents/files. It takes lots of time in calculating, hence thecost and calculating time increases drastically. Proposed algorithm makes single pass over the data points, hence no need to calculate same points again and again. It draw a simple sketch of number of clusters which is in greater number than actual cluster. Then ballkmeans algorithm is applied to get the actual number of cluster.

D. Mini Batch K-Means Algorithm

It is assumed that by using one-pass over the input data and produces as many centroids as it determines is optimal can reduce time drastically. Avoiding multiple passes over the input data can have major impacts on running time because just reading large data set can increase the cost in large-scale computations. The basic idea behind this algorithm is that incoming points are assigned to a nearby cluster or used as basis of new cluster. Decision is made according to how distance to nearby cluster compares to adaptive scale parameter, then the limitations of existing system can be overcome. Here clusters are formed by assigning each data points to the nearest centroids. First step is processed at Map function, runs by Mapper and second step is processed by Reduce function, runs by Reducer.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 2, Issue 2, pp: (336-342), Month: April-June 2014, Available at: www.researchpublish.com

Mini Batch K-Means algorithm steps as follows:

1. *The cluster mapper function*

Input: Observations x1, x2, x3, x4….xn, initial distancecutoff fo, target number of centroids ko, distance d.

Output: distance cutoff f, centroids C = {c1, c2, c3,…..ck}

Let c be the closest cluster to point p

Let d be the distance between c and p

If d > distanceCutoff, create a new cluster from p

Else if d <= distanceCutoff, create a new cluster with probability d/distanceCutoff

Else merge p into c

2. *The cluster reducer function*

Input: distanceCutoffs and centroids from each mapper {(f1, c1)……(fr, cr)}, target final number of centroids k

Output: Centroids (ci) where i = 1….ko

Clusters formed assigning each data point to nearest centroid

Center of mass of trimmed clusters become new centroids (ci)

IV. EXPERINMENTS AND RESULTS

The implementation phase of any system development is the most important phase as it yields the solution, which solves the problem at hand. In implementation stage theoretical design created in the design phase is converted into a working system. It is the most critical stage in development of a new system because it involves study of the existing system, careful planning, constraints on implementation, designing of methods to make changes, and evaluation of newly created system. Implementation have been performed by taking more than one data sets and results have been compare between k-Means and proposed system Mini Batch K-Means.

A.Implementation Environment

K-means is implemented under the following environments:

1) Hadoop 1.0.2;
2) Java 7;
3) Nodes;
4) VMware for creating Nodes;
5) Mahout

TABLE I: NODES DETAILS

Node Id	CPU	Memory	Operating System
Master-Node	Intel i5 – 450M	1.00 GB	Ubuntu 12.04
Slave-Node2	Intel i5 – 450M	512.00 MB	Ubuntu 12.04
Slave-Node3	Intel i5 – 450M	512.00 MB	Ubuntu 12.04
Slave-Node4	Intel i5 – 450M	512.00 MB	Ubuntu 12.04
Secondary Name-Node	Intel i5 – 450M	512.00 MB	Ubuntu 12.04

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 2, Issue 2, pp: (336-342), Month: April-June 2014, Available at: www.researchpublish.com

B.Data-Sets

- *Reuters21578 dataset*

The documents in the Reuters-21578 is mainly a collection appeared on the Reuters newswire in 1987. The documents are keeps to be assembled and indexed with categories by personnel from Reuters Ltd. The Reuters-21578 collection is probably distributed in 22 files. Each of the first 21 files (reut2-000.sgm through reut2-020.sgm) contains 1000 documents, while the last (reut2-021.sgm) contains 578 documents. The files are in SGML format and it is available on UCI machine learning repository.

- *SC time series dataset*

This dataset contains 600 examples of control charts synthetically generated by the process in Alcock and Manolopoulos (1999). There are six different classes of control charts: Normal Cyclic, Increasing trend, Decreasing trend, upward shift, downward shift of columns.

C. Result Comparison

- Using Reuters21578 dataset

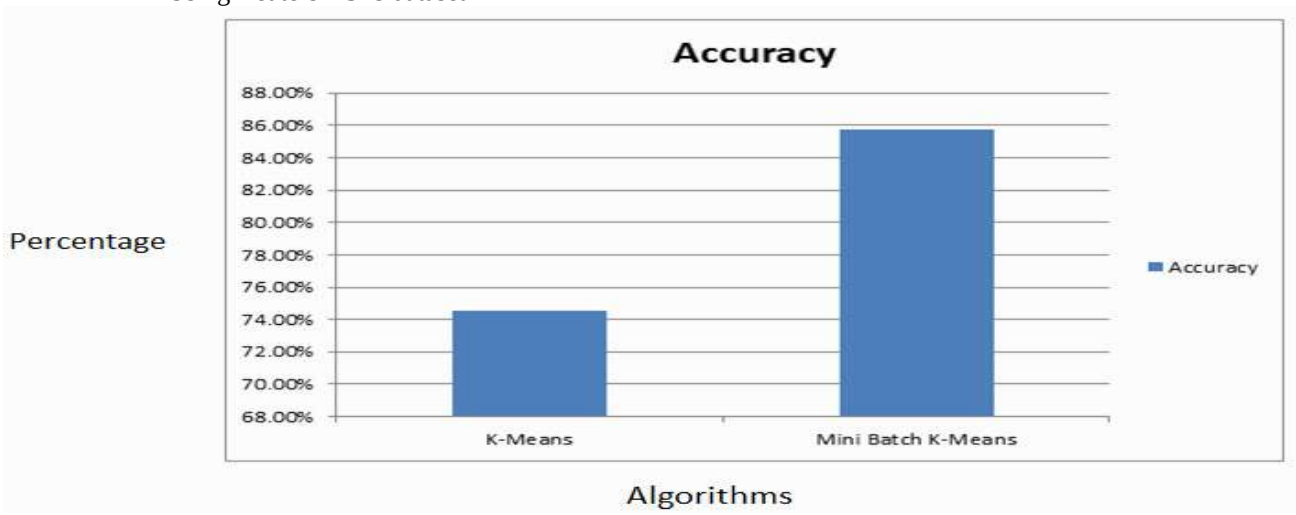

Fig 1 Accuracy comparison using Reuters21578 dataset

The results obtained by using Mini Batch K-Means and K-Means algorithm are compared by taking Reuters21578 data-set. It can clearly be seen that Mini Batch K-Means form better clustering as its accuracy is having high percentage.

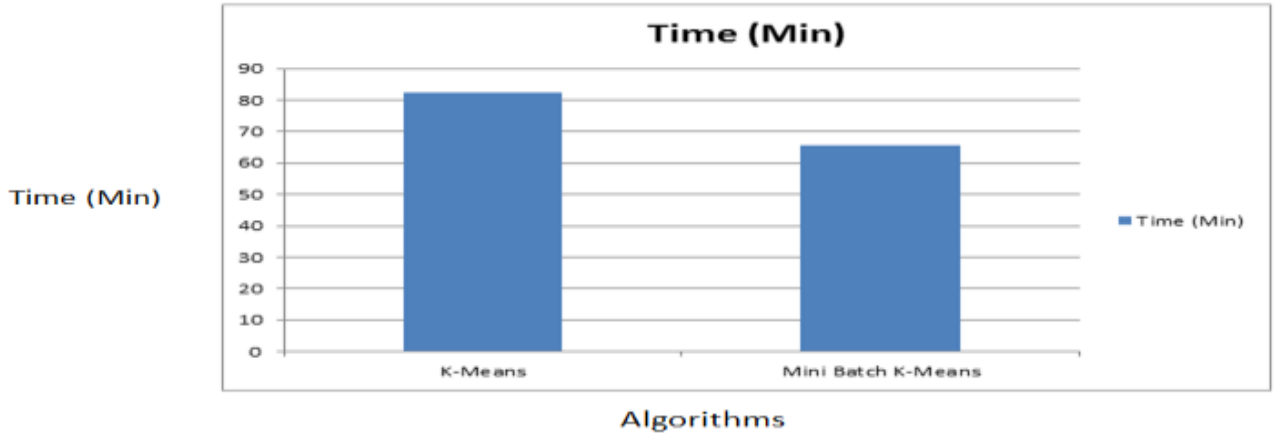

Fig 2Time taken by algorithms performing clustering

The results obtained by using Mini Batch K-Means and K-Means algorithm are compared by taking Reuters21578 data-set. It can clearly be seen that Mini Batch K-Means takes less time in obtaining clustering.

- Using SC time series dataset

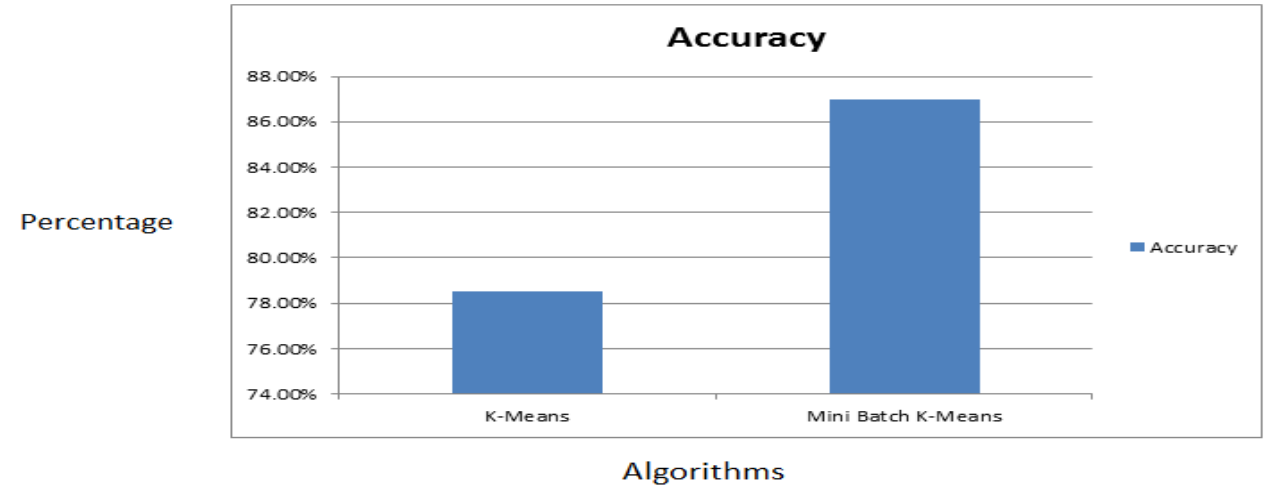

Fig 3Accuracy comparison using SC time series Data-set

The results obtained by using Mini Batch K-Means and K-Means algorithm are compared by taking SC time series data-set. It can clearly be seen that Mini Batch K-Means form better clustering as its accuracy is having high percentage.

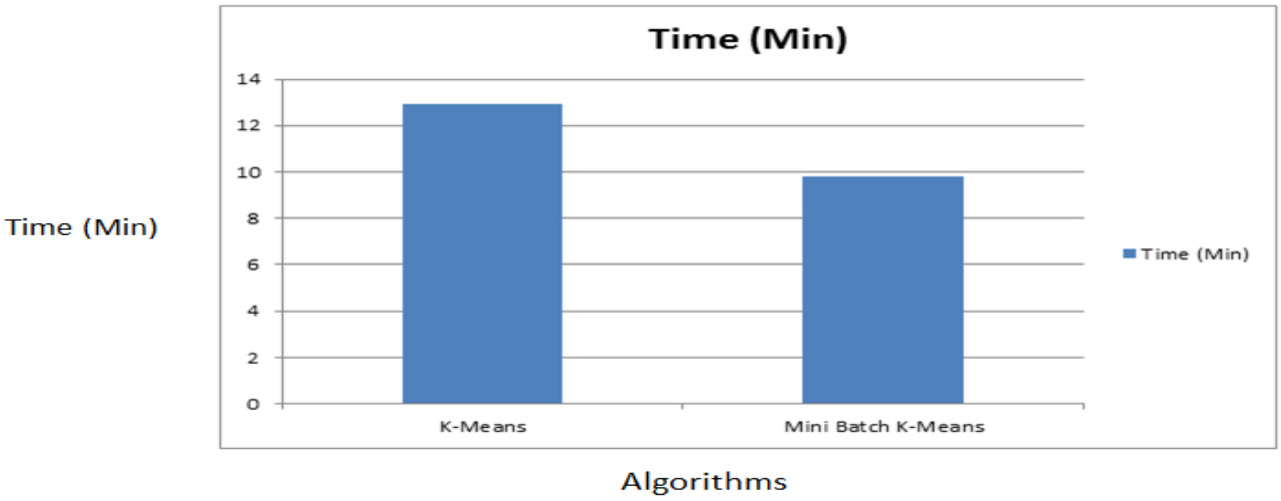

Fig 4Time taken by algorithms performing clustering

The results obtained by using Mini Batch K-Means and K-Means algorithm are compared by taking SC time series data-set. It can clearly be seen that Mini Batch K-Means takes less time in obtaining clustering.

V. CONCLUSION

The volume of information exchange in today's world engages huge quantity of data processing. Existing clustering algorithm like K-Means, where randomly centroids are selected spend most of the time by calculating distance between the center and data. Implementation of proposed algorithm Mini Batch K-Means over a distributed network by using Map-Reduce, not only provide a robust and efficient system for grouping of data with similar characteristics but also reduces the implementation costs of processing such huge volumes of data. Mini Batch K-Means can increase reduce computation time and provide better accuracy by dealing with various dataset. Mini Batch K-Means uses single pass over data, hence there is no need to read the same point many times and compare its distance with each centroids at each iteration.

Streaming and online generating data-set can used this centroid updating and cluster forming technique to decrease computation time of cluster and to have better accuracy. Twitter, facebook, you-tube etc. can use this technique in future to find which people belong to what kind of activity and their interest.

REFERENCES

[1] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung. "The Google File System." Proceedings of the nineteenth ACM symposium on Operating systems principles. , pp. 29-43, ACM, New York, NY, USA, 2003.

[2] Jeffrey Dean, and Sanjay Ghemawat. "MapReduce: Simplified Data Processing on Large Clusters." Proceedings of the 6th conference on Symposium on OperatingSystems Design & Implementation., pp. 137-150, USENIX Association, Berkley,CA, USA, 2004.

[3] C C Chang, B He, Z Zhang. Mining semantics for large scale integration on the web: evidences, insights, and challenges. SIGKDD Explorations, 2004: 6(2):67-76

[4] J. Coldberger, S. Gordon, H. Greenspan, "Unsupervised image-set clustering using an information theoretic framework," IEEE transactions on image processing, vol.15, no. 2, pp. 449-457, 2006

[5] A. McCallum, K. Nigam, and L. H. Ungar. Efficient Clustering of High Di-mensional Data Sets with Application to Reference Matching. Association for Computing Machinery (ACM), New York, NY, USA, 2000.

[6] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu. An Efficient k-Means Clustering Algorithm: Analysis and Implemen-tation. IEEE Computer Society, Washington, DC, USA, 2002

[7] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wal-lach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber.

[8] "Bigtable: A Distributed Storage System for Structured Data." ACM Transac-tions on Computer Systems, vol. 26(4), ACM, New York, NY, USA, 2008.

[9] R. De Maesschalck, D. Jouan-Rimbaud, and D.L. Massart The Mahalanobis distance. Texas A&M University, College Station, TX, USA, 2000.

[10] So What is Hadoop? (2010). Retrieved February 2011, from Atbrox: http://atbrox.com/2010/02/17/hadoop/

[11] How Map and Reduce Operations are Actually Carried Out. (2009).Retrieved February 2011, from Hadoop Wiki: http://wiki.apache.org/hadoop/HadoopMapReduce

[12] Google and IBM Announce University Initiative to Address Internet-Scale Computing Challenges. (2007). Retrieved February 2011, from IBM: http://www-03.ibm.com/press/us/en/pressrelease/

[13] Apache Hadoop. http://hadoop.apache.org/.

[14] T. White, "Hadoop, the Definitive Guide," O'Reilly Media, May 2009.

[15] A. Arimond. A Distributed System for Pattern Recognition andMachine Learning. Master's thesis, TU Kaiserslautern and DFKI, 2010