Computer and Information Science; Vol. 8, No. 4; 2015
ISSN 1913-8989 E-ISSN 1913-8997
Published by Canadian Center of Science and Education

Model of Cloud-Based Services for Data Mining Analysis

Aleksandar Karadimce[1], Slobodan Kalajdziski[1] & Danco Davcev[1]

[1] Faculty of Computer Science and Engineering, University Ss Cyril and Methodius, Skopje, Macedonia

Correspondence: Aleksandar Karadimce, Faculty of Computer Science and Engineering, University Ss Cyril and Methodius, Skopje, Macedonia. Tel: 389-70-300-787. E-mail: akaradimce@ieee.org

Received: August 20, 2015 Accepted: August 27, 2015 Online Published: November 6, 2015

doi:10.5539/cis.v8n4p40 URL: http://dx.doi.org/10.5539/cis.v8n4p40

Abstract

New cloud-based services are being developed constantly in order to meet the need for faster, reliable and scalable methods for knowledge discovery. The major benefit of the cloud-based services is the efficient execution of heavy computation algorithms in the cloud simply by using Big Data storage and processing platforms. Therefore, we have proposed a model that provides data mining techniques as cloud-based services that are available to users on their demand. The widely known data mining algorithms have been implemented as Map/Reduce jobs that are been executed as services in cloud architecture. The user simply chooses or uploads the dataset to the cloud, makes appropriate settings for the data mining algorithm, executes the job request to be processed and receives the results. The major benefit of this model of cloud-based services is the efficient execution of heavy computation data mining algorithm in the cloud simply by using the Ankus - Open Source Big Data Mining Tool and StarfishHadoop Log Analyzer. The expected outcome of this research is to offer the integration of the cloud-based services for data mining analysis in order to provide researchers with reliable collaborative data mining analysis model.

Keywords: data mining, cloud computing services, Map/Reduce, web services, knowledge discovery

1. Introduction

Researchers and scientists during their activities have a constant need to conduct data mining analysis on the gathered or updated dataset. The researcher simply chooses appropriate e-service, inserts personal data to the web-form and lets the platform do the data processing in order to deliver the results. Therefore, we have proposed a platform that provides e-services that are available to users on their demand. This activity requires to have installed appropriate software for data mining, run the analysis and wait for the results to come up. As the dataset gets bigger the analysis takes more time to get the results of the data mining analysis. In order to facilitate the data mining analysis process, particular researchers have proposed a cloud-based platform that provide data mining as a service (DMaaS) (Chen et al., 2012). The backend of the data processing engine is Hadoop, an open-source implementation of the MapReduce Framework (Velusamy et al., 2013) and the implementation of the data mining algorithms is done in Apache Mahout (Walunj and Sadafale, 2013). The Hadoop systems are using HDFS (Hadoop Distributed File System) infrastructure data cluster file system that stores, processes, retrieves and manages data, as mentioned by Shvachko et al. (2010). The performance of individual machine learning algorithms within each cloud computing framework, such as Apache Mahout or GraphLab frameworks remains mainly unknown (Li et al., 2013). Authors Li et al. (2013) have advised on the absence of robust selection methodology for matching the input data with effective machine learning algorithms, which limits the practitioners to make effective use of cloud computing. Therefore, the focus of this research is on developing a model of cloud-based services for data mining analysis that will have intuitive and easy to use GUI.

The aim of this research is to propose a model of cloud-based services that will provide the user with a complete analysis of the completed Hadoop job. This will allow us to use the processing power of Hadoop and develop both advanced and nation-wide scalable data mining algorithms for researchers with the specific requirements. Also, the proposed cloud-based service will enable a user with a web-based platform for running the data mining analysis. The main contribution of this paper is to demonstrate the performance analysis of running data mining algorithm and the easy integration of cloud-based services. The results of the conducted experiments will verify the proposed aim of this research paper.

This paper is organized as follows: Section II gives an overview of the related work. Section III describes the architecture of the model of cloud-based services for data mining analysis. Section IV addresses the implementation of data mining algorithms in Map/Reduce jobs. Section V presents our experimental results and discussion. Finally, Section VI concludes the paper and proposes future work.

2. Related Work

Researchers challenge nowadays is not the complexity of the problems to be solved, but the amount of data to be taken into consideration when doing it. In order to solve these data problems while solving the inherent problems of distributed computing at the same time, a radically new programming model with parallel execution on the cluster, called Map/Reduce (Dean & Ghemawat, 2008), is needed. The Map/Reduce framework can simplify the complexity of running distributed data processing for large-scale data-intensive applications, such as data mining (Xie et al., 2010).

The simplest and the oldest kind of recommendation is editorial or hand curated recommendation system. We might find a list of favorites, for example when we go into a bookstore, we might find staff picks, and on certain websites we can see a list of staff favorites or a list of essential items. These are essentially built by hand. These are products that have been picked by the editorial staff to feature on the home page. If we go beyond editorial recommendations, the next simple thing that we can do is simple aggregates. On many websites, well see lists of top ten, or most popular, or most recent for example, if we go to YouTube we can see the most popular videos, for instance. So these are simple aggregates which sort of taking into account user activity to make recommendations to other users. But these recommendations don't depend on the user they only depend on the aggregate activity of a lot of other users. The third kind of recommendation is recommendations that are tailored to individual users. For example, book recommendations tailored to our taste, movie recommendations based on the movies that we watched previously or music recommendation based on our music interests. Existing research proves that the single-node machines that have implemented a recommender algorithm are time-consuming and they are unable to meet the computing needs of larger data sets (Lin et al., 2014). To improve the performance of the algorithm, they have proposed a distributed collaborative filtering recommendation algorithm combining k-means and slope one on Hadoop (Lin et al., 2014).

The e-commerce systems have key challenges for recommender systems in which high-quality recommendations are required and more recommendations per second for millions of customers and products need to be performed (Jiang et al., 2011). There have been developed item-based collaborative filtering algorithms that are based on Map/Reduce, by splitting the three most costly computations in the proposed algorithm into four Map-Reduce phases, each of which can be independently executed on different nodes in parallel (Jiang et al., 2011).

Authors Walunj and Sadafale (2013) have proposed the Mahout as highly scalable solution, which is able to support distributed processing of large data sets across clusters of computers using the Hadoop. The Mahout data mining libraries have been used for predictive analytics in the context of text classification, recommender systems, and decision support systems (Hammond & Varde, 2013). With the integration of Mahout Machine Learning algorithms into the cloud-based framework for real-world industrial applications, researchers are able to do anomaly detection, fault mode prediction (Xu et al., 2013) and optimization (Drre et al., 2015). These related solutions have the main challenge to present efficient data mining systems that are able to scale up their data interpretation abilities to discover interesting patterns in large datasets (Pavlo et al., 2009). Most of the proposed solutions require extensive preparation of the researchers working environment. Therefore, we have detected the emergence of data mining as a service (DMaaS) (Chen et al., 2012), which is able to provide a cloud-based data mining platform. In order to have efficient and complete data mining analysis, we have proposed the following architecture of the model based on cloud services.

3. Architecture of the Model of Cloud-Based Services for Data Mining Analysis

The main purpose of the model of cloud-based services for data mining is to provide an integrated environment where researchers will receive complete analysis. This section gives an elaborate description of the proposed architecture of cloud-based services for conducting data mining analysis. The cloud-based service platform for running the data mining analysis is Ankus - Open Source Big Data Mining Tool (Ren et al., 2014). Ankus platform allows integration with the HDFS Cluster file system and the Web Server, see Figure 1. The client will be able to access the cloud-based services for data mining analysis by means of a web browser. The successful executing the Map/Reduce job generates output folder with results and another folder with log history of the executed job. The information stored in history folder consisted of job configuration file and java .jar file is been used to run performance analysis of the executed job. After successful execution of Map/Reduce job researcher receives the results and can make profounder performance analysis using the StarfishHadoop Log Analyzer (Herodotou et al., 2011). This Analyzer can be run locally on the client's computer or remotely in the cloud, which is represented with a dashed line in Figure 1.

Firstly, the researchers are authenticated using username and password on the web-based platform for running the data mining analysis, see Figure 2-(1). The next task is a researcher to choose or upload the dataset to the cloud by using the GUI navigated HDFS file system, see Figure 2-(2). The dataset from researchers computer is been transferred to the client application, which writes an HDFS file. Then it first splits the file into HDFS blocks and the Name Node receives a list of Data Nodes which are replicas (3 copies) of each block and writing data is done by multithreading (Kala Karun and Chitharanjan, 2013). After writing down the dataset to the HDFS file system the user simply, makes appropriate settings depending on the type of the data mining algorithm that should be executed. The process of executing the data mining algorithm is based on Map/Reduce parallel processing model.

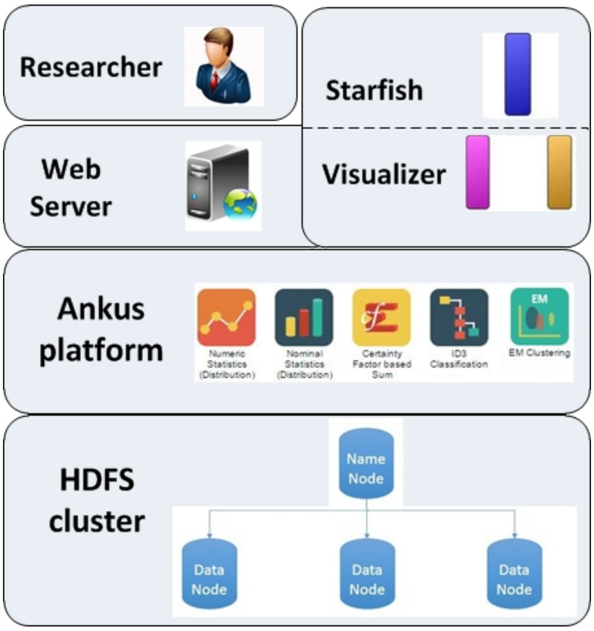
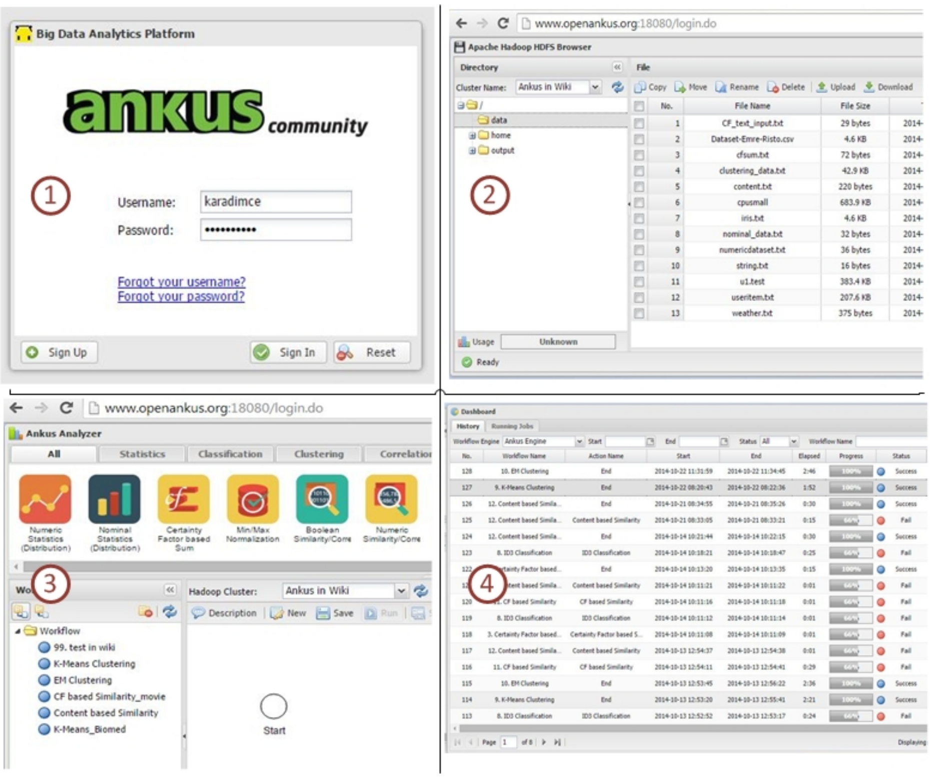


Figure 1. Architecture of the model of cloud-based services for data mining analysis

Map/Reduce is a model for processing key, value-based data in parallel, and consists of two steps of carrying out the Map task on the basis of input data sources to create interim results, and carrying out the Reduce task by using the interim results as input to obtain final results (Dean and Ghemawat, 2008). The Mapper function transforms a key and a value to a list of key-value pairs, as mentioned by Dean and Ghemawat (2008). On the other side, the Reducer function transforms its parameters a key and a list of values to a list of key-value pairs (Dean and Ghemawat, 2008). For the most common data mining algorithms that have been implemented as Map/Reduce jobs, detailed explanation is provided in Section III of this paper. The data redundancy is achieved by using HDFS cluster file system, and the data parallelism is realized by executing the same function on each constituting piece of data that is (implicitly) processed.

Researchers can use the Analyzer flow in a drag and drop manner to run data mining algorithms with custom settings, see Figure 2-(3). The complete information of the Map/Reduce job processing status is followed on Web Server Dashboard, see Figure 2-(4). Here researchers can find detailed information for user configuration settings and error log information for the failed jobs.



www.openankus.org:18080/login.do

Username: karadimce

Password: ••••••••••

Cluster Name: Ankus in Wiki

No.	File Name	File Size	
1	CF_text_input.txt	29 bytes	2014-
2	Dataset-Emre-Risto.csv	4.6 KB	2014-
3	cfsum.txt	72 bytes	2014-
4	clustering_data.txt	42.9 KB	2014-
5	content.txt	220 bytes	2014-
6	cpusmall	683.9 KB	2014-
7	iris.txt	4.6 KB	2014-
8	nominal_data.txt	32 bytes	2014-
9	numericdataset.txt	36 bytes	2014-
10	string.txt	16 bytes	2014-
11	u1.test	383.4 KB	2014-
12	useritem.txt	207.6 KB	2014-
13	weather.txt	375 bytes	2014-

Hadoop Cluster: Ankus in Wiki

Workflow
- 99. test in wiki
- K-Means Clustering
- EM Clustering
- CF based Similarity_movie
- Content based Similarity
- K-Means_Biomed

Workflow Engine: Ankus Engine Start: End: Status: All Workflow Name:

No.	Workflow Name	Action Name	Start	End	Elapsed	Progress	Status
128	10. EM Clustering	End	2014-10-22 11:31:59	2014-10-22 11:34:45	2:46	100%	● Success
127	9. K-Means Clustering	End	2014-10-22 08:20:43	2014-10-22 08:22:36	1:52	100%	● Success
126	12. Content based Simila...	End	2014-10-21 08:34:55	2014-10-21 08:35:26	0:30	100%	● Success
125	12. Content based Simila...	Content based Similarity	2014-10-21 08:33:05	2014-10-21 08:33:21	0:15	66%	● Fail
124	12. Content based Simila...	End	2014-10-14 10:21:44	2014-10-14 10:22:15	0:30	100%	● Success
123	8. ID3 Classification	ID3 Classification	2014-10-14 10:18:21	2014-10-14 10:18:47	0:25	66%	● Fail
122	...rtainty Factor based...	End	2014-10-14 10:13:20	2014-10-14 10:13:35	0:15	100%	● Success
12[illegible]	...tent based Simila...	Content based Similarity	2014-10-14 10:11:21	2014-10-14 10:11:22	0:01	66%	● Fail
120	...1. CF based Similarity	CF based Similarity	2014-10-14 10:11:16	2014-10-14 10:11:18	0:01	66%	● Fail
119	8. ID3 Classification	ID3 Classification	2014-10-14 10:11:12	2014-10-14 10:11:14	0:01	66%	● Fail
118	3. Certainty Factor based...	Certainty Factor based S...	2014-10-14 10:11:08	2014-10-14 10:11:09	0:01	66%	● Fail
117	12. Content based Simila...	Content based Similarity	2014-10-13 12:54:37	2014-10-13 12:54:38	0:01	66%	● Fail
116	11. CF based Similarity	CF based Similarity	2014-10-13 12:54:11	2014-10-13 12:54:41	0:29	66%	● Fail
115	10. EM Clustering	End	2014-10-13 12:53:45	2014-10-13 12:56:22	2:36	100%	● Success
114	9. K-Means Clustering	End	2014-10-13 12:53:20	2014-10-13 12:55:41	2:21	100%	● Success
113	8. ID3 Classification	ID3 Classification	2014-10-13 12:52:52	2014-10-13 12:53:17	0:24	66%	● Fail

Figure 2. Demonstration of using the cloud-based services for data mining analysis

4. Implementation of Data Mining Algorithms in Map/Reduce Jobs

Considering that our intention was that researchers to be able to analyze a variety of dataset more easily, simple and fast, therefore, we have proposed to use web-based environment. Each type of the well-known data mining and machine learning analysis algorithms has a different implementation in order to be executed as Map/Reduce job.

4.1 Statistics Algorithms

From the statistics algorithms, we have represented the implementation of the Numeric Statistics and Nominal Statistics. The statistics algorithms are implemented in Map/Reduce as two single Map/Reduce Jobs, visually described in Figure 3.

4.1.1 Numeric Statistics

The numeric statistics analyzer is calculating a set of vector-based statistics for the numerical data stated in the data file.

a) Step 1 Map/Reduce: Divides the data for the distributed processing with a manifold of n (the number of reducers in Hadoop system). Calculates a value for each of basic statistics and does the final summing.

b) Step 2 Map/Reduce: After the first step the basic statistics are calculated, it conducts overall summing by the n-fold in order to calculate the overall base statistics. The output is the ordered vector-based field with these properties: index, sum, mean, harmonic mean, geometric mean, variance, standard deviation, maximum, minimum, median.

4.1.2 Nominal Statistics

The nominal statistics analyzer is calculating a vector of categorical data in a file-based statistics (frequency and percentage). For each categorical attribute of the input dataset, the nominal statistics is calculating the frequency and percentage of each attribute type. For example, if the input is iris dataset (Ren et al., 2014), then the output data file contains a calculation of the frequency and percentage of each type of iris flower.

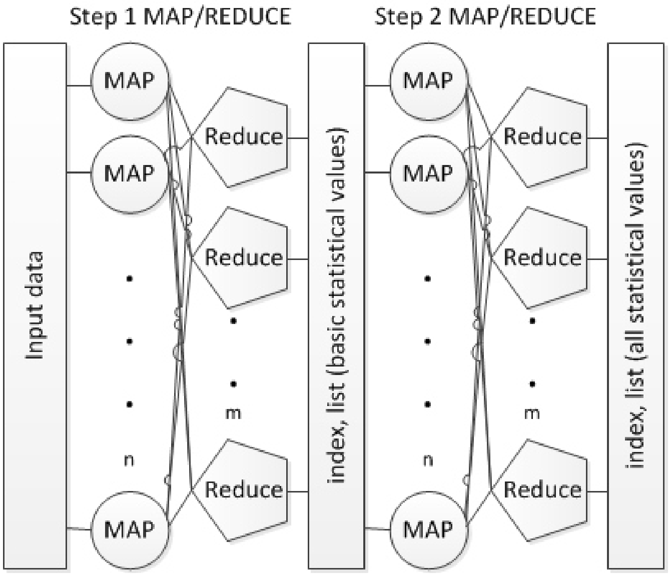


Figure 3. Implementation of Numerical statistics as Map/Reduces job

Step 1 Map/Reduce: Calculates the frequency of categorical attributes.

Step 2 Map/Reduce: First is reusing the total number of Map performed in step 1 to obtain the total number of data, and it calculates the ratio of each category of the property by using the second Map/Reduce step.

4.2 Clustering Algorithms

Clustering algorithms are the most commonly used data mining method to determine the relationship between objects. We have chosen to demonstrate the EM Clustering algorithm.

4.2.1 EM Clustering Algorithm

The EM-based Clustering is been conducted on the vector-based data files, which supports both numeric and categorical attributes. It uses distribution with mean and variance as a percentage of the distance group and uses frequency of categorical attributes and crowded center of the category. The EM Clustering is been implemented as Map/Reduce process as follows (see Figure 4):

Step 1 Map/Reduce: The Mapper is determining the center of the initial clusters. Initially, it calculates the probability that of random belonging to each cluster to all data.

Step 2 Map/Reduce: In this step there is an assignment to the community center and community updates.

Step 2-1: (Map processes of the Map/Reduce job) clustering assignment. For each data, the value is been calculated the high probability of belonging to their own communities.

Step 2-2: (Reduce process of Map/Reduce jobs) community center update. The value is been assigned per cluster that belongs, based on this data there is an update on the center of the cluster.

Step 2-3: termination condition determines. If the center of the updated cluster is same as the center of the old cluster it ends the Step-2 Map/Reduce. Otherwise, there is re-done of the Step-2 Map/Reduce.

The process of clustering ends if it exceeds the number set by the input parameter (max_iterations). This condition is checked after the Map/Reduce Step-2, see Figure 4.

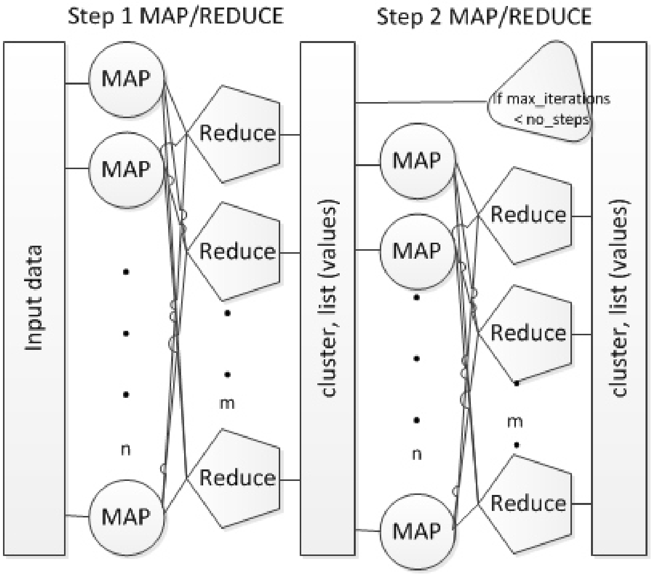


Figure 4. Implementation of Clustering algorithm as Map/Reduce job

4.2.2 K-means Clustering Algorithm

Similarly like the EM Clustering, the k-means clustering algorithm can be implemented as Map/Reduce process. It uses the Euclidean distance to calculate the distance between the center and crowded item numerical properties (Ren et al., 2014).

4.3 Recommendation Algorithms

The main idea behind content-based recommendation systems is to recommend items to a customer similar to previous items rated highly by the same customer. The content-based approach is able to deal marginally with the fact the users can have unique tastes as long as we can build item profiles for the items that the user likes first. Considering that our intention is to use the content-based recommendation system by exploiting the cloud services. We have proposed to be used the following implementation of recommendation algorithm as Map/Reduce job.

4.3.1 Content-Based Similarity

The content-based similarity module analyzes the content of a content-based similarity recommendation service. The similarity can be calculated using the dice coefficient, Jaccard coefficient or hamming distance. Usually, we calculate the content-based similarity for recommendation data sets that are consisted of the three columns (userid, item, rating) sequentially into the columns.

- First column: userid
- Second columns: item
- Third column: rating

Similarly like the content based similarity, the user based recommendation and item based recommendation algorithms can be implemented as Map/Reduce process. The content-based recommendation similarity is this implemented in Map/Reduce job and the process is as follows (see Figure 5):

Step 1 Map/Reduce: The input data file comes under the unique key value as parameters. The Mapper is grouping the values by column unique key values. The Reducer rearranges the values for the similarity / distance analysis and returns a column with the unique key value received by the i-th iteration value set.

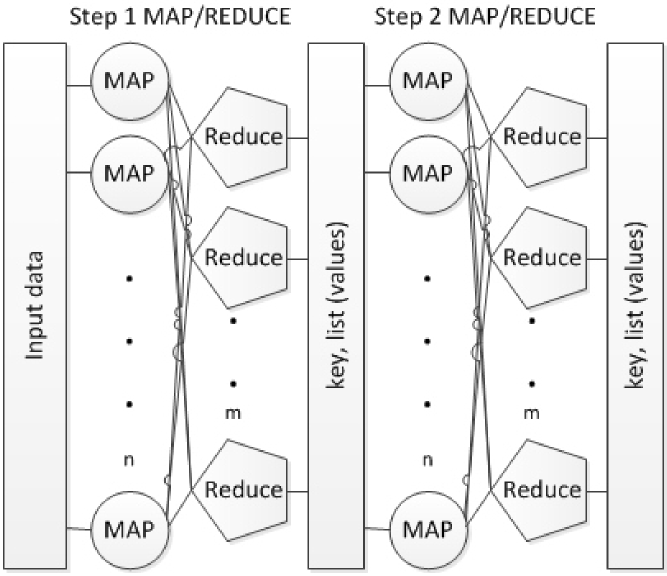


Figure 5. Implementation of content based Recommendation algorithm as Map/Reduce job

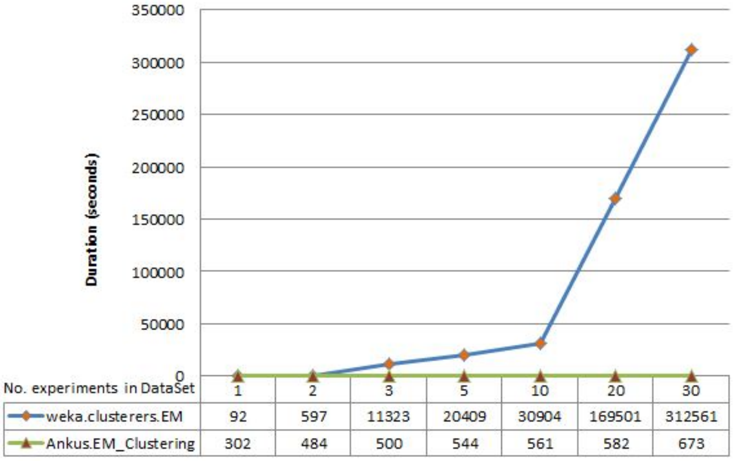
Step 2 Map/Reduce: The Mapper is rearranging the values and prepares them for the conditional similarity/ distance calculation. The Reducer returns the column with the unique key value received by the i-th iteration value set with the result for each calculation.

5. Experimental Results and Discussion

The proposed model of cloud-based services for data mining analysis has been supported with an experiment done using large data set. We have chosen a dataset of experiments in DMB format from Institute of Systems Analysis and Informatics "Antonio Ruberti" (Ruberti, 2015). The first row of the dataset file contains the names of the experiments. The first column contains the feature name (the variable of the experiments) and the second column describes the variable type: NUM (numerical values). The experiments for data mining analysis have been done using the clustering algorithm. This algorithm has been selected because it needs a large amount of computational time for handling large data sets (Sewisy et al., 2014). As a referential tool for the conducted experiments we have used WEKA (Hall et al., 2009), a tool for automatic learning and data mining algorithms, similar researches based on WEKA have been done (Srivastava, 2014; Rupali et al., 2014).

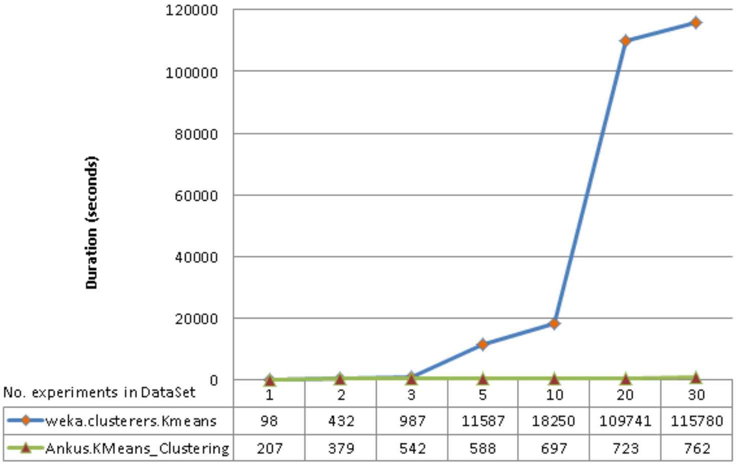
The first experiment was comprised of a comparison of the performance results received from the WEKA and the proposed model of cloud-based services for data mining. We have used the same clustering algorithm EM-clustering in both cases. Comparison of the received performance results running the EM-clustering algorithm with WEKA (run on the single machine) and proposed cloud-based services for data mining (run on HDFS cluster) is shown in Figure 6.

The comparison has shown that for a smaller number of experiments in dataset there is no significant difference in duration of processing time. As the number of experiments in dataset is increasing it significantly increases the time duration (seconds) to execute the EM-clustering using the WEKA tool. Nevertheless, the duration to execute the EM-clustering using the proposed model of cloud-based services for data mining is not taking more than 10 minutes regardless of the increase of a number of experiments of dataset. The most significant performance improvement is noticed when there are considered more than 10 experiments of dataset to be executed the EM-clustering algorithm.



No. experiments in DataSet	1	2	3	5	10	20	30
weka.clusterers.EM	92	597	11323	20409	30904	169501	312561
Ankus.EM_Clustering	302	484	500	544	561	582	673

Figure 6. Demonstration of using the cloud-based services for data mining analysis (EM-clustering)



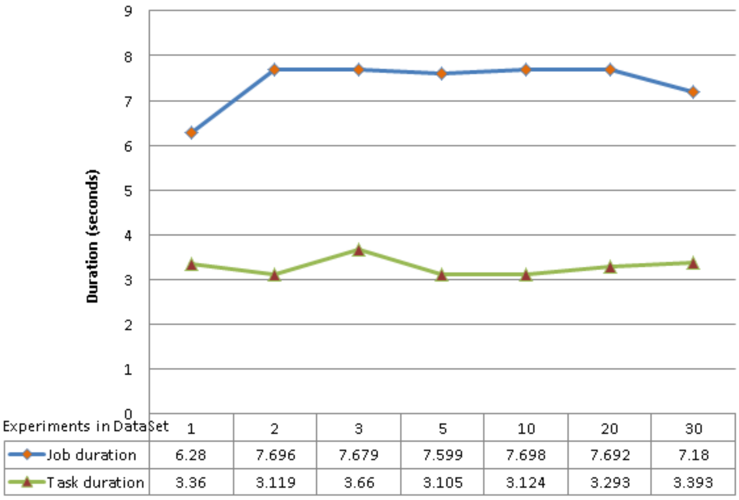
No. experiments in DataSet	1	2	3	5	10	20	30
weka.clusterers.Kmeans	98	432	987	11587	18250	109741	115780
Ankus.KMeans_Clustering	207	379	542	588	697	723	762

Figure 7. Demonstration of using the cloud-based services for data mining analysis (k-means clustering)

The second experiment was conducted on the same dataset by using the k-means clustering algorithm. The algorithm has provided better performance results for the same dataset, see Figure 7. We have noticed a significant difference in duration of processing time when more than 10 experiments of dataset have been executed with the k-means clustering algorithm.

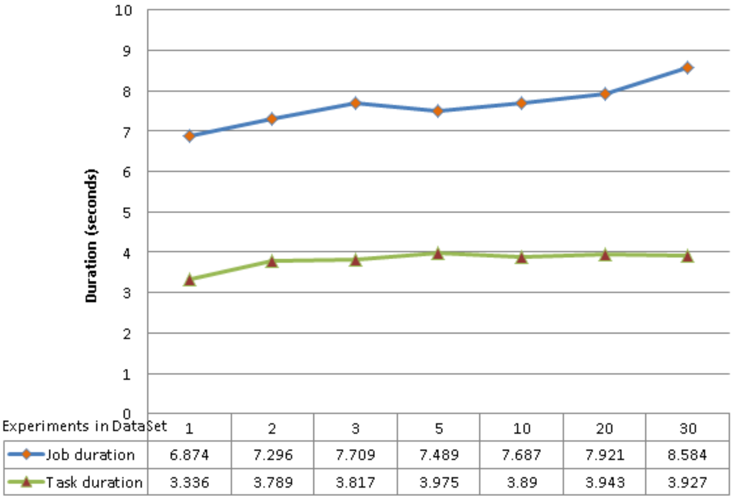
For a complete analysis of the finished Hadoop jobs, we have analyzed the Job duration and Task duration using the StarfishHadoop Log Analyzer. The timeline analysis of the completed Map/Reduce jobs has provided more detailed information. The process of building of service-oriented model requires service providers to build an integrated application platform.

For the first experiment that has been done with the EM-clustering algorithm, the Job duration took twice more time than the Task duration. The duration trend line has been increasing for Job duration except for the last Map/Reduce execution when the number of experiments is 30. In that case, the Job duration was faster than the previous executions (see Figure 8). This means that Map/Reduce jobs had a normal duration; only the time needed for the Task has been shorter for half second.



Experiments in DataSet	1	2	3	5	10	20	30
Job duration	6.28	7.696	7.679	7.599	7.698	7.692	7.18
Task duration	3.36	3.119	3.66	3.105	3.124	3.293	3.393

Figure 8. Comparison of MapReduce Task duration and job duration using services for data mining analysis (EM-clustering)



Experiments in DataSet	1	2	3	5	10	20	30
Job duration	6.874	7.296	7.709	7.489	7.687	7.921	8.584
Task duration	3.336	3.789	3.817	3.975	3.89	3.943	3.927

Figure 9. Comparison of MapReduce Task duration and job duration using services for data mining analysis (k-means clustering)

Comparison of the Job duration and Task duration, for the execution of Map Reduce jobs, for the second experiment that has been done with the k-means clustering algorithm and is given in Figure 9. There we have not observed any significant changes in duration trend line for Job duration and Task duration. The Task duration trend-line is following the Job duration, with duration values twice as bigger measured in seconds.

The main advantage of the proposed model of using cloud-based services is the efficient execution of heavy computation data mining algorithm by using the Map Reduce concept. The GUI interface provided by Ankus - Open Source Big Data Mining Tool gives improved navigation for data mining analysis and very natural integration for visualizing the performance results using the StarfishHadoop Log Analyzer. This platform is responsible for gathering all context related information; with proper selection of cloud-based services we will be able to deliver the users with context-aware information.

The cloud-based services offer reliable, faster and collaborative data mining analysis model, because the results of every analysis are saved in the cloud. Model is every convenient for research purposes because a scientist can repeat the experiments and easily compare the gathered performance results of data mining analysis. The experiments have demonstrated improved performance characteristic with a comparison to WEKA data mining tool.

6. Conclusion and Future Work

In this paper has been proposed a model of cloud-based services for data mining analysis. The proposed model has provided researchers with integrated cloud-based services for data mining analysis, which has intuitive and easy to use GUI. Clustering algorithms has been chosen for evaluation because they need a large amount of computational time for handling large data set. Based on the completed experiments it has been confirmed the performance benefit of using the model of cloud-based services for data mining analysis compared to the traditional data mining tools.

The main purpose of the model of cloud services platform is to provide an integrated environment where research institutions will receive complete data analysis. The proposed model has provided integration of the cloud-based services for data mining analysis that can be used by researchers for reliable collaborative data mining analysis. The clustering algorithm EM-clustering was used for comparison of the performance results of data mining analysis. The biggest difference in duration of processing time was experienced when more than 10 experiments of the dataset have been executed.

Future investigation in the field of cloud-based services will continue with the analysis of social networking datasets. Running data mining analysis on these datasets will bring interesting facts for the user important social behavior. Cloud-based services are facing the privacy risk because they collect sensitive user's records. The main issue is that this information is sent to a remote storage location for processing. This procedure raises the security issues because there is a possible risk of exposing user's data by the unreliable third party mobile cloud service vendor. The risks are expected to be overcome with the introduction of standardization in the field of cloud computing technology.

References

Chen, T., Chen, J., & Zhou, B. (2012). A system for parallel data mining service on cloud, *Second International Conference on Cloud and Green Computing*, CGC 2012, Xiangtan, Hunan, China, pp. 329–330. http://dx.doi.org/10.1109/CGC.2012.49

Dean, J., & Ghemawat, S. (2008). Mapreduce: Simplified data processing on large clusters. *Communications of the ACM - 50th-anniversary issue: 1958 – 2008, 51*(1), 107–113. http://doi.acm.org/10.1145/1327452.1327492

Dörre, J., Apel, S., & Lengauer, C. (2015). Modeling and optimizing mapreduce programs. *Concurrency and Computation*: *Practice Experience, 27*(7), 1734–1766. http://dx.doi.org/10.1002/cpe.3333

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The weka data mining software: An update. *ACM SIGKDD Explorations Newsletter, New York, NY, USA, 11*(1), 10–18. http://dx.doi.org/10.1145/1656274.1656278

Hammond, K., & Varde, A. S. (2013). Cloud-based predictive analytics: Text classification, recommender systems and decision support. *Proceedings of the IEEE 13th International Conference on Data Mining Workshops, ICDMW 2013, Dallas, TX, USA*, pp. 607–612. http://doi.ieeecomputersociety.org/10.1109/ICDMW.2013.95

Herodotou, H., Lim, H., Luo, G., Borisov, N., Dong, L., Cetin, F. B., & Babu, S. (2011). Starfish: A self-tuning system for big data analytics, *Proceedings of the 5th Conference on Innovative Data Systems Research*, CIDR 2011, Asilomar, California, USA, pp. 261–272.

Jiang, J., Lu, J., Zhang, G., & Long, G. (2011). Scaling-up item-based collaborative filtering recommendation algorithm based on hadoop, *Proceedings of the IEEE World Congress on Services, SERVICES 2011, Washington, DC, USA,* pp. 490–497. http://doi.ieeecomputersociety.org/10.1109/SERVICES.2011.66

Kala Karun, A., & Chitharanjan, K. (2013). A review on hadoop - hdfs infrastructure extensions, *Proceedings of the IEEE Conference on Information Communication Technologies, ICT 2013, JeJu Island*, pp. 132–137. http://dx.doi.org/10.1109/CICT.2013.6558077

Li, K., Gibson, C., Ho, D., Zhou, Q., Kim, J., Buhisi, O., Brown, D. E., & Gerber, M. (2013). Assessment of machine learning algorithms in cloud computing frameworks, *Proceedings of the IEEE Systems and Information Engineering Design Symposium, SIEDS 2013, Charlottesville, VA, USA,* pp. 98–103. http://10.1109/SIEDS.2013.6549501

Lin, K., Wang, J., & Wang, M. (2014). A hybrid recommendation algorithm based on hadoop, *Proceedings of the 9th International Conference on Computer Science Education, ICCSE 2014, Vancouver, BC, Canada,* pp. 540–543. http://dx.doi.org/10.1109/ICCSE.2014.6926520

Pavlo, A., Paulson, E., Rasin, A., Abadi, D. J., DeWitt, D. J., Madden, S., & Stonebraker, M. (2009). A comparison of approaches to large-scale data analysis, Proceedings of the 2009 ACM SIGMOD *International Conference on Management of Data, New York, NY, USA,* pp. 165–178. http://doi.acm.org/10.1145/1559845.1559865

Ren, Z., Wan, J., Shi, W., Xu, X., & Zhou, M. (2014). Workload analysis, implications, and optimization on a production hadoop cluster: A case study on taobao. *IEEE Transactions on Services Computing, 7*(2), 307-321. http://doi.ieeecomputersociety.org/10.1109/TSC.2013.40

Ruberti, A. (2015). *Istituto di analisideisistemie informatica Antonio Ruberti data set on experiments conducted in the institute*. Retrieved June 21, 2015, from http://dmb.iasi.cnr.it/inputexamples/example.csv

Rupali, B., Kalbhor, M., Shinde, S., & Rajeswari, K. (2014). Improvement of expectation maximization clustering using select attribute. *International Journal of Computer Science and Mobile Computing(IJCSMC), 3*(4), 503–508.

Sewisy, A. A., Marghny, M. H., AbdElAziz, R. M., & Taloba, A. I. (2014). Fast efficient clustering algorithm for balanced data, *International Journal of Advanced Computer Science and Applications, 5*(6), 123–129. http://dx.doi.org/10.14569/IJACSA.2014.050619

Shvachko, K., Kuang, H., Radia, S., & Chansler, R. (2010). The hadoop distributed file system, *Proceedings of the IEEE 26th Symposium on Mass Storage Systems and Technologies, MSST 2010, Incline Village, NV, USA,* pp. 1–10. http://dx.doi.org/10.1109/MSST.2010.5496972

Srivastava, S. (2014). Weka: A tool for data preprocessing, classification, ensemble, clustering and association rule mining. *International Journal of Computer Applications*, *88*(10), 26–29. http://dx.doi.org/10.5120/15389-3809

Velusamy, K., Venkitaramanan, D., Vijayaraju, N., Suresh, G., & Madhu, D. (2013). Inverted indexing in big data using hadoop multiple node cluster. *International Journal of Advanced Computer Science and Applications, 4*(11), 156–161. http://dx.doi.org/10.14569/IJACSA.2013.041122

Walunj, S. G., & Sadafale, K. (2013). An online recommendation system for e-commerce based on apache mahout framework. *Proceedings of the 2013 Annual Conference on Computers and People Research, New York, NY, USA,* pp. 153–158. http://dx.doi.org/10.1145/2487294.2487328

Xie, J., Yin, S., Ruan, X., Ding, Z., Tian, Y., Majors, J., Manzanares, A., & Qin, X. (2010). Improving mapreduce performance through data placement in heterogeneous hadoop clusters, *Proceedings of the IEEE International Symposium on Parallel Distributed Processing, Workshops and Phd Forum, IPDPSW 2010, Charlottesville, VA, USA,* pp. 1–9. http://dx.doi.org/10.1109/IPDPSW.2010.5470880

Xu, B., Mylaraswamy, D., & Dietrich, P. (2013). A cloud computing framework with machine learning algorithms for industrial applications. *Proceedings of the International Conference on Artificial Intelligence 2013, Las Ve- gas, NV, USA.*

Copyrights