International Journal of
Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-5, Issue-4)

Research Article | **April 2016**

A System for Image Conversion Using Mapreduce in Cloud Computing Environment

Vinit Mohata, Prof. Dhananjay M. Dakhane, Prof. Ravindra L. Pardhi
Department of Computer Science and Engineering, Amravati University,
Maharashtra, India

Abstract—

Cloud computing is a colloquial expression used to describe a variety of different types of computing concepts that involve a large number of computers connected through a real-time communication network. Many service providers who release Social Network Services (SNS) [3] allow users to disseminate multimedia objects. SNS and media content providers are constantly working toward providing multimedia-rich experiences to end users [4]. In order to develop a SNS based on large amounts of social media, scalable mass storage for social media data created daily by users is needed. Although the ability to share multimedia objects makes the Internet more attractive to consumers, clients and underlying networks are not always able to keep up with this growing demand. Here, we apply a cloud computing environment to our Hadoop-based Image transcoding system. Improvements in quality and speed are achieved by adopting Hadoop Distributed File System (HDFS) [12] for storing large amounts of image data created by numerous users, MapReduce [10] for distributed and parallel processing of image data.

Keywords— Cloud computing, HDFS, cloud infrastructure, Image Conversion, Big Data.

I. INTRODUCTION

Cloud computing [1] has achieved remarkable interest from researchers and the IT industry for providing a flexible dynamic IT infrastructure, QoS guaranteed computing environments, and configurable software services [2]. Due to these advantages, many service providers who release Social Network Services (SNS) [3] allows users to disseminate multimedia objects. SNS and media content providers are constantly working toward providing multimedia-rich experiences to end users [4]. In order to develop a SNS based on large amounts of social media, scalable mass storage for social media data created daily by users is needed. Although the ability to share multimedia objects makes the Internet more attractive to consumers, clients and underlying networks are not always able to keep up with this growing demand. Multimedia processing is characterized by large amounts of data, requiring large amounts of processing, storage, and communication resources, thereby imposing a considerable burden on the computing infrastructure. The traditional approach to transcoding multimedia data requires specific and expensive hardware because of the high-capacity and high definition features of multimedia data. Therefore, general purpose devices and methods are not cost effective, and they have limitations. Here, we apply a cloud computing environment to our Hadoop-based Image Conversion system. Improvements in quality and speed are achieved by adopting Hadoop Distributed File System (HDFS) [12] for storing large amounts of image data created by numerous users, MapReduce [10] for distributed and parallel processing of image data. This platform is composed of two parts: A social media data analysis platform for large scalable data analysis; a cloud distributed and parallel data processing platform for storing, distributing, and processing social media data.

II. ANALYSIS OF PROBLEM

Current approaches to processing images depend on processing a small number of images having a sequential processing nature. These processing loads can almost fit on a single computer equipped with a relatively small memory. Still, we can observe that more disk space is needed to store the large-scale image repository that usually results from satellite-collected data.

The current processing of images goes through ordinary sequential ways to accomplish this job. The program loads image after image, processing each image alone before writing the newly processed image on a storage device. Generally, we use very ordinary tools that can be found in Photoshop, for example. Besides, many ordinary C and Java programs can be downloaded from the Internet or easily developed to perform such image processing tasks. Most of these tools run on a single computer with a Windows operating system. Although batch processing can be found in these single-processor programs, there will be problems with the processing due to limited capabilities.

With the proliferation of online photo storage and social medias from websites such as Facebook, Flickr, and Picasa, the amount of image data available is larger than ever before and growing more rapidly every day [18]. This alone provides an incredible database of images that can scale up to billions of images. Incredible statistical and probabilistic models can be built from such a large sample source. The management of unstructured data is recognized as one of the major unsolved problems in the information technology (IT) industry, the main reason being that the tools and techniques that have proved so successful transforming structured data into business intelligence and actionable information simply don't work when it comes to unstructured data. Unstructured data files often include text and multimedia content.

© 2016, IJERMT All Rights Reserved

Examples include e-mail messages, word processing documents, videos, photos, audio files, presentations, webpages and many other kinds of business documents. Note that while these sorts of files may have an internal structure, they are still considered "unstructured" because the data they contain doesn't fit neatly in a database. Experts estimate that 80 to 90 percent of the data in any organization is unstructured. And the amount of unstructured data in enterprises is growing significantly -- often many times faster than structured databases are growing. New approaches are necessary. Therefore, we are in need of a new parallel approach to work effectively on massed image data. In order to process a large number of images effectively, we use the Hadoop HDFS to store a large amount of remote sensing image data, and we use MapReduce to process these in parallel. The MapReduce programming model will be actively working with this distributed file system.

It is these reasons that motivate the need for research with vision applications that take advantage of large sets of images. Hadoop can process stores of both unstructured and structured data that are extremely large, very complex and changing rapidly.

HDFS is characterized as a highly fault-tolerant distributed file system that can store a large number of very large files on cluster nodes. MapReduce provides an extremely powerful framework that works well on data-intensive applications where the model for data processing is similar or the same. It is often the case with image-based operations that we perform similar operations throughout an input set, making MapReduce ideal for image-based applications. However, many researchers find it impractical to be able to collect a meaningful set of images relevant to their studies [19]. Additionally, many researchers do not have efficient ways to store and access such a set of images. As a result, little research has been performed on extremely large image-sets. The goal of this image transcoding is to create a tool that will make development of large-scale image processing.

III. PROPOSED WORK

In our proposed work we are trying to make an application which uses Hadoop MapReduce and Cloud Environment. Which help to build an effective system for processing of big data processing and storage.

Hadoop it is a Flexible Infrastructure for large scale computation and data processing on a network of commodity hardware. HDFS is the primary storage system used by Hadoop applications [5]. HDFS creates multiple replicas of data blocks and distributes them on computed nodes throughout a cluster to enable reliable and extremely rapid computations. HDFS has a master-slave structure and uses the TCP/IP protocol to communicate with each node. MapReduce is a programming model for the parallel processing of distributed large-scale data [6]. MapReduce processes an entire large-scale data set by dividing it among multiple servers.

MapReduce frameworks provide a specific programming model and a run-time system for processing and creating large amounts of datasets which is amenable to various real-world tasks [8]. MapReduce framework also handles automatic scheduling, communication, synchronization for processing huge datasets and it has the ability related with fault tolerance. MapReduce programming model is executed in two main steps, called mapping and reducing. Mapping and reducing are defined by mapper and reducer functions that are data processing functions. Each phase has a list of key and values pairs as input and output. In the mapping, MapReduce input datasets and then feeds each data element to the mapper as a form of key and value pairs. In the reducing, all the outputs from the mapper are processed and a final result is created by reducer with merging process. MapReduce frameworks provide a specific programming model and a run-time system for processing and creating large amounts of datasets which is amenable to various real-world tasks [8]. MapReduce framework also handles automatic scheduling, communication, synchronization for processing huge datasets and it has the ability related with fault tolerance. MapReduce programming model is executed in two main steps, called mapping and reducing. Mapping and reducing are defined by mapper and reducer functions that are data processing functions. Each phase has a list of key and values pairs as input and output. In the mapping, MapReduce input datasets and then feeds each data element to the mapper as a form of key and value pairs. In the reducing, all the outputs from the mapper are processed and a final result is created by reducer with merging process.

For making effective Image conversion application we have to utilize Hadoop MapReduce, cloud environment in single system. Which will help to solve the problems of Big data compression and limited storage. Hadoop MapReduce includes several stages, each with an important set of operations helping to get to your goal of getting the answers you need from Big Image data. The process starts with a user request to run a MapReduce program and continues until the results are written back to the HDFS. HDFS and MapReduce perform their work on nodes in a cluster hosted on racks of commodity servers. By using Hadoop, MapReduce, Cloud we can solve all our problems related to big data processing and storage.

IV. SYSTEM MODULES

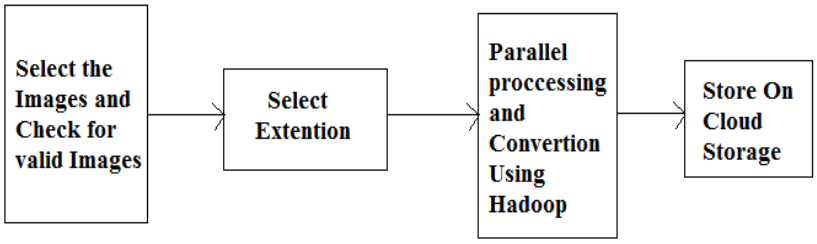

Fig.1 Image Conversion Modules

© 2016, IJERMT All Rights Reserved

1. Image Selection:

First Module of the system is Image selection. In image selection we are selecting a folder of images which we are converting. Only the valid images are selected from the folder for further processing. Valid images are the images which are valid image type such as jpg, bmp, tiff, png, etc.

2. Select Extension:

In the second module we can select the image extension for the selected images folder. For now further processing we are now enter the conversion criteria as per our requirements for image conversion. Quality and division factor to be entering as per requirements.

3. Image Conversion:

After entering the criteria for image conversion, now the image conversion is started on the selected images. The image conversion is done in such way that multiple images are converting in parallel by which time required to convert to images is decreases as compare to normal process. Here we are using open library Hadoop Image Processing Interface (HIPI) of Hadoop for image conversion. HIPI Provide an open, extendible library for image processing and computer vision applications in a MapReduce framework. HIPI Store images efficiently for use in MapReduce applications. HIPI Allow for simple filtering of a set of images.

4. Store images on cloud:

Images which are processed are now stored on the cloud storage to minimize the large data storage problem.

V. EVALUATION

The privileged server used in the experiments for evaluation is a single enterprise scale cluster. Table 1 lists the specifications of the evaluation cluster. Because the structure of the cluster is homogeneous, it provides a uniform evaluation environment. We have evaluate the process in the following configuration.

Table I. Evaluation Cluster Specifications

CPU	Intel core i5 2.6 GHz
RAM	4GB DDR 3
HDD	500GB
OS	Windows 10
Java	1.7 jdk
Library	HIPI- Hadoop image processing interface
Eclipse	Eclipse – Mars

Five data sets were used to verify the performance of the proposed module. Table II lists the specific information about the data sets used. During the experiment, the following default options in Hadoop were used. We evaluated the processing Time of the proposed module and optimized it. We planned and executed the experiments. In the First experiment, we calculate the processing time of proposed module with a non-Hadoop-based system and the second experiment we calculate the processing time with Hadoop system. We measured each running time taken in our server using only sequential programing using HIPI libraries, respectively.

Table II. Input Output Specification

Data Size(MB)	**Total Images**	**Source Format**	**Destination Format**
1	8	JPG	JPG,BMP,PNG
2	14	JPG	JPG,BMP,PNG
5	34	JPG	JPG,BMP,PNG
10	62	JPG	JPG,BMP,PNG
25	129	JPG	JPG

We provided the following data as input and converted into the target format as shown in table II data specification after the execution the system module also we have studied and plotted the graph as shown in figure 2.

© 2016, IJERMT All Rights Reserved

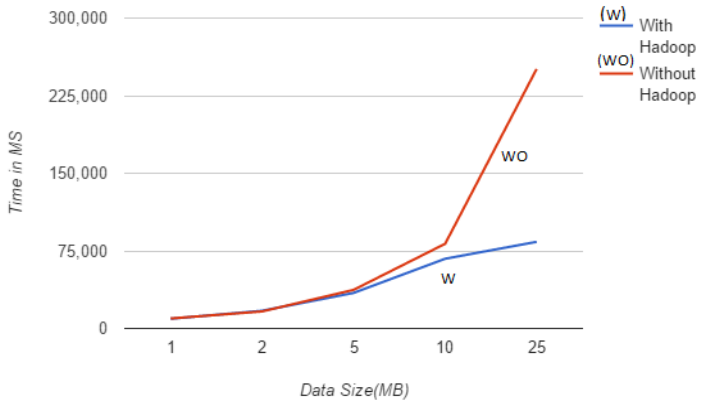

Fig.2 Image Conversion Evaluation

The processing time for the sequential processing without Hadoop always required large as compared to MapReduce when the data is too large. However, after 62 files, the difference between the performances of the two experiments grows when the number of processing files exceeds a certain level; the task of creating Map generates JVM overhead. The option of reusing JVM is a possible solution to reduce overhead created by processing numerous small files on HDFS, as can be seen in the results presented above also we reduce the Burdon of computing power because we processed the data without any hardware, the whole system are designed in JAVA and HIPI library using HDFS.

VI. APPLICATIONS

1. **Social Media:** The Large data is generated from the social media platforms such as YouTube, Facebook, Twitter, LinkedIn, and Flickr. The amount of images being uploaded to the internet is rapidly increasing, with Facebook users uploading over 2.5 billion new photos every month. It can be used to improve applications performance by greatly reducing the file size and network bandwidth required to display your application.
2. **Business Applications:** online shopping application where the every item has image data is shown. Company's employee data and scan copies of various documents.
3. **Satellite images:** This includes weather data or the data that the government captures in its satellite surveillance imagery.
4. **Photographs and video:** This includes security, surveillance, and traffic video.

VII. CONCLUSION

In this System, we have made a MapReduce based image conversion module in a cloud-computing environment to solve the problem of computing infrastructure overhead. In our experiment we conclude that for the larger image data processing Hadoop based system works more effectively than the normal systems. As we know image data processing is time consuming and requires large computing resources so using Hadoop based image conversion we can solve both issues. We implemented an image conversing module that exploits the advantages of cloud computing.

REFERENCES

[1] M. Kim and H. Lee, "SMCC: Social media cloud computing model for developing SNS based on social media," Communications in Computer and Information Science, vol.206, pp.259-266, 2011.

[2] Z. Lei, "Media transcoding for pervasive computing," in Proc. of 5th ACM Conf. on Multimedia, no4, pp.459-460, Oct. 2001.

[3] Sun-Moo Kang, Bu-Ihl Kim, Hyun-Sok Lee, Young-so Cho, Jae-Sup Lee, Byeong-Nam Yoon, "A study on a public multimedia service provisioning architecture for enterprise networks", Network Operations and Management Symposium, 1998, NOMS 98., IEEE, 15-20 Feb 1998,44-48 vol.1, ISBN : 0 -7803-4351-4.

[4] Hyeokju Lee, Myoungjin Kim, Joon Her, and Hanku Lee, "Implementation of MapReduce-based Image Conversion Module in Cloud Computing Environment," International Conference on Information Networking (icoin) 2012 .Date 1-3 Feb. 2012.

[5] R. Buyya, C. Yeo, S. Venugopal, J. Broberg, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Computer Systems, vol.25, no.6, pp.599-616, Jun. 2009.

[6] G. Barlas, "Cluster-based optimized parallel video transcoding," Parallel Computing, vol.38,no.4-5, pp.226-244, Apr. 2012.

[7] Myoungjin Kim, Seungho Han, Yun Cui, Hanku Lee, and Changsung Jeong," A Hadoop-based Multimedia Transcoding System for Processing Social Media in the PaaS Platform of SMCCSE," KSII Transaction on Internet and Information Systems VOL. 6, NO. 11, Nov 2012.

[8] I. Ahmad, X. Wei, Y. Sun and Y.-Q. Zhang," Video transcoding: An overview of various techniques and research issues," IEEE Transactions on Multimedia, vol.7, no.5, pp.793-804 Oct. 2005.

[9] S. Ghemawat, H. Gobioff and S.-T. Leung, "The google file system," Operating Systems Review (ACM), vol.37, no.5, pp.29-43, Oct. 2003.

© 2016, IJERMT All Rights Reserved

[10] D. Seo, J. Kim and I. Jung, "Load distribution algorithm based on transcoding time estimation for distributed transcoding servers," in Proc. of 2010 Conf. on Information Science and Applications, article no.5480586, Apr. 2010.

[11] D.M. Boyd and N.B. Ellison, "Social network sites: Definition, history, and scholarship," Journal of Computer-Mediated Communication, vol.13, no.1, pp.210-230, Oct. 2007.

[12] J. Shafer, S. Rixner and A.L. Cox, "The Hadoop distributed file system: Balancing portability and performance," in Proc. of IEEE International Symposium on Performance Analysis of Systems and Software, pp.122-133, Mar. 2010.

[13] Hadoop MapReduce project, http://hadoop.apache.org/mapreduce/

[14] Hari Kalva, Aleksandar Colic, Garcia, Borko Furht, "Parallel programming for multimedia applications", Multimedia Tools and Applications, volume 51, number 2, 901-818, DOI:10.1007/s11042-010-o656-2.

[15] H. Kocakulak and T. T. Temizel, "A Hadoop solution for ballistic image analysis and recognition," in 2011 Int. Conf. High Performance Computing and Simulation (HPCS), Istanbul, pp. 836–842.

[16] B. Li, H. Zhao, Z. H. Lv, "Parallel ISODATA clustering of remote sensing images based on MapReduce," in 2010 Int. Conf. Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Huangshan, pp. 380–383.

[17] N. Golpayegani and M. Halem, "Cloud computing for satellite data processing on high end compute clusters," in Proc. 2009 IEEE Int. Conf. Cloud Computing (CLOUD '09), Bangalore, pp. 88–92.

[18] FACEBOOK, 2010. Facebook image storage. http://blog.facebook.com/blog.php?post=206178097130.

© 2016, IJERMT All Rights Reserved