Marozzo, F., Belcastro, L. (2022). Cloud Computing for Big Data Analysis. In: Zomaya, A., Taheri, J., Sakr, S. (eds) Encyclopedia of Big Data Technologies. Springer, Cham. https://doi.org/10.1007/978-3-319-63962-8_136-2

Cloud Computing for Big Data Analysis

Fabrizio Marozzo, Loris Belcastro

Definitions

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction (Mell and Grance (2011)).

Overview

In the last years the ability to produce and gather data has increased exponentially. In fact, in the Internet of Things' era, huge amounts of digital data are generated by and collected from several sources, such as sensors, cams, in-vehicle infotainment, smart meters, mobile devices, GPS devices, web applications and services. Moreover, thanks to the growth of social networks (e.g., Facebook, Twitter, Pinterest, Instagram, Foursquare, etc.) and the widespread diffusion of mobile phones every day millions of people share information about their interests and activities. Because of their characteristics, such huge volumes of data, commonly referred as Big Data, represent a challenge to the current storage, process and analysis capabilities. The information contained in such data is of great value for industry and science. Consequently, many researches are focusing on the development of technologies for extracting value from this data in a reasonable time. To overcame Big Data issues and get valuable information and knowledge in shorter time, high performance and scalable computing systems are used in combination with data and knowledge discovery techniques.

In this context, Cloud computing has emerged as an effective platform to face the challenge of extracting knowledge from Big Data repositories in limited

time, as well as to provide an effective and efficient data analysis environment for researchers and companies. From a client perspective, the Cloud is an abstraction for remote, infinitely scalable provisioning of computation and storage resources (Talia et al (2015)). The National Institute of Standards and Technology (NIST) provided the following definition Cloud computing: "*A model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*". From the NIST definition, we can identify five essential characteristics of Cloud computing systems, which are: *i*) on-demand self-service, *ii*) broad network access, *iii*) resource pooling, *iv*) rapid elasticity, and *v*) measured service. According to the above definition, all the computing resources, dynamically scalable and often virtualized, are allocated on-demand somewhere "in the Cloud" and provided as services over the Internet.

Cloud computing vendors provide their services according to three main distribution models:

- *Software as a Service (SaaS)*, in which software and data are provided through Internet to customers as ready-to-use services. Specifically, software and associated data are hosted by providers, and customers access them without need to use any additional hardware or software.
- *Platform as a Service (PaaS)*, in an environment including databases, application servers, development environment for building, testing and running custom applications. Developers can just focus on deploying of applications since Cloud providers are in charge of maintenance and optimization of the environment and underlying infrastructure.
- *Infrastructure as a Service (IaaS)*, that is an outsourcing model under which customers rent resources (e.g., CPUs, disks, virtualized servers). Compared to the other models, IaaS has higher system administration costs for the user, but it allows a full customization of the execution environment.

Key Research Findings

Most available Cloud-based data analysis systems today are based on open source frameworks, such as Hadoop[1] and Spark[2], but there are also some proprietary solutions proposed by big companies (e.g., IBM, Kognitio). To cope with the need of processing Big Data, such systems should meet some requirements (Belcastro et al (2019a)):

- *Efficient data management and exchange*. Big Data sets are often arranged by gathering data from several heterogeneous and sometimes not well-known sources. In this context, data processing systems must support efficient protocols for data transfers and for communications as well as they have to enable local computation of data sources and fusion mechanisms to compose the results produced in distributed nodes.
- *Interoperability*. It is a main issue in large-scale applications that use re-

[1] https://hadoop.apache.org/

[2] https://spark.apache.org/

sources such as data and computing nodes. systems for Big Data should support interoperability by allowing the use of different data formats and tools.
- *Efficient parallel computation*. An effective approach for analyzing large volumes of data and obtaining results in a reasonable time is based on the exploitation of inherent parallelism of the most data analysis and mining algorithms. Thus, systems for Big Data analysis have to allow parallel data processing and provide a way to easy monitor and tune the degree of parallelism.
- *Scalability*. With the exponential increases in the volume of data to be processed, systems for Big Data processing must accommodate rapid changes in the growth of data, either in traffic or volume, by exploiting the increment of computational or storage resources efficiently.

Concerning the distribution models of Cloud services, the most common ones for providing Big Data analysis solutions are PaaS and SaaS. Usually, IaaS is not used for high-level data analysis applications but mainly to handle the storage and computing needs of data analysis processes.

With the PaaS model users do not need to care about configuring and scaling the infrastructure (e.g., a distributed and scalable Hadoop system), because the Cloud vendor will do that for them. Finally, the SaaS model is used to offer complete Big Data analysis applications to end users, so that they can execute analysis on large and/or complex datasets by exploiting Cloud scalability in storing and processing data.

As outlined in Li et al (2010), users can access Cloud computing services using different client devices, Web browsers and desktop/mobile applications. Several technologies and standards are used by the different components of the architecture. For example, users can interact with Cloud services through SOAP-based or RESTful Web services (Richardson and Ruby (2008)) and Ajax technologies, which let Cloud services to have look and interactivity equivalent to those provided by desktop applications.

Developing Cloud-based Big Data analysis applications may be a complex task, with specific issues that go beyond those of stand-alone application programming. For instance, Cloud programming must deal with deployment, scalability and monitoring aspects that are not easy to handle without the use of ad-hoc environments (Talia et al (2015)). In fact, to simplify the development of Cloud applications, specific development environments are often used. Some of the most representative Cloud computing development environments currently in use can be classified into four types:

- *Integrated development environments*, which are used to code, debug, deploy and monitor Cloud applications that are executed on a Cloud infrastructure, such as Eclipse, Visual Studio and IntelliJ.
- *Parallel-processing development environments*, which are used to define parallel applications for processing large amount of data that are run on a cluster of virtual machines provided by a Cloud infrastructure (e.g., Hadoop and Spark).
- *Workflow development environments*, which are used to define workflow-based applications that are executed

on a Cloud infrastructure, such as Swift and DMCF.
- *Data-analytics development environments*, which are used to define data analysis applications through machine learning and data mining tools provided by a Cloud infrastructure. Some examples are Azure ML and BigML.

The programming model is a key factor to be considered for exploiting the powerful features of Cloud computing. In the last years several programming models have been proposed for exploiting the potential of Cloud computing and addressing the challenge posed by Big Data, such as MapReduce, workflows, message passing, bulk synchronous parallel. Systems that implement such models can be compared according to four criteria for assessing their suitability for parallel programming (Belcastro et al (2019a)): *i*) *level of abstraction* that refers the programming capabilities of hiding low-level details of a system; *ii*) *type of parallelism* that describes the way in which a system allows to express parallel operations; *iii*) *infrastructure scale* that refers to the capability of a system to efficiently execute applications taking advantage from the infrastructure size; and *iv*) *classes of applications* that describes the most common application domain of a system.

MapReduce (Dean and Ghemawat (2004)) is widely recognized as one of the most important programming models for Cloud computing environments, being it supported by Google and other leading Cloud providers such as Amazon, with its Elastic MapReduce service, and Microsoft, with its HDInsight, or on top of private Cloud infrastructures such as OpenNebula.

Apache Hadoop is the most popular open source implementation of MapReduce. It can be adopted for developing distributed and parallel applications using many programming languages. Hadoop relieves developers from having to deal with classical distributed computing issues, such as load balancing, fault tolerance, data locality, and network bandwidth saving. The Hadoop project is not only about the MapReduce programming model, as it has become a reference for several other programming systems, such as: Storm and Flink for streaming data analysis; Giraph and Hama for graph analysis; Pig and Hive for querying large datasets. The Hadoop-ecosystem is undoubtedly one of the most complete solution for data analysis problem, but at the same time it is thought for high skilled users.

On the other hand, many other solutions are designed for low-skilled users or for low-medium organizations that do not want to spend resources in developing and maintaining enterprise data analysis solutions. Two representative examples of such data analysis solutions are Microsoft Azure Machine Learning and Data Mining Cloud Framework.

Microsoft Azure Machine Learning (Azure ML)[3] is a SaaS for the creation of machine learning workflows. It provides a very high-level of abstraction, because a programmer can easily design and execute data analytics applications by using simple drag-and-drop web interface and exploiting many built-in tools for data manipulation and machine learning algorithms.

The Data Mining Cloud Framework (DMCF) (Marozzo et al (2015)) is a software system developed at University

[3] https://azure.microsoft.com/services/machine-learning-studio/

of Calabria for allowing users to design and execute data analysis workflows on Clouds. DMCF supports a large variety of data analysis processes, including single-task applications, parameter sweeping applications, and workflow-based applications. A workflow in DMCF can be developed using a visual- or a script-based language. The visual language, called VL4Cloud (Marozzo et al (2016)), is based on a design approach for end users having a limited knowledge of programming paradigms. The script-based language, called JS4Cloud (Marozzo et al (2015)), provides a flexible programming paradigm for skilled users who prefer to code their workflows through scripts. VL4Cloud/JS4Cloud workflows can also include MapReduce-based workflows that are executed in parallel on DMCF enabling scalable data processing on Clouds (Belcastro et al (2015)).

Other solutions have been created mainly for scientific research purposes and, for this reason, they are poorly used for developing business applications (e.g., E-Science Central, COMPSs, and Sector/Sphere).

E-Science Central (e-SC) (Hiden et al (2013)) is a Cloud-based system that allows scientists to store, analyze and share data in the Cloud. It provides a user interface that allows programming visual workflows in any Web browser.

e-SC is commonly used to provide a data analysis back end to standalone desktop or Web applications. To this end, the e-SC API provides a set of workflow control methods and data structures. In the current implementation, all the workflow services within a single invocation of a workflow execute on the same Cloud node.

COMPSs (Lordan et al (2014)) is a programming model and execution runtime which aims to ease the development of parallel applications for distributed infrastructures, such as clusters and Clouds. With COMPSs, users create a sequential application and specify which methods of the application code will be executed remotely. Providing an annotated interface where these methods are declared with some metadata about them and their parameters does this selection. The runtime intercepts any call to a selected method creating a representative task and finding the data dependencies with all the previous ones that must be considered along the application run. A new system built on top of COMPSs, namely PyCOMPSs (Tejedor et al (2017)), has been also proposed with the aim of facilitate the development of parallel applications in Python for distributed infrastructures.

Sector/Sphere (Gu and Grossman (2009)) is an open source Cloud framework designed to implement data analysis applications involving large, geographically distributed datasets. The framework includes its own storage and compute services, called Sector and Sphere respectively, which allow to manage large dataset with high performance and reliability.

Examples of Application

Cloud computing has been used in many scientific fields, such as astronomy, meteorology, social computing, and bioinformatics, which are greatly based on scientific analysis on large volume of data. In many cases, developing and configuring Cloud-based applications

requires an high level of expertise, which is a common bottleneck in the adoption of such applications by scientists.

Many solutions for Big Data analysis on Clouds have been proposed in bioinformatics, such as: SparkSeq (Wiewiórka et al (2014)) is a Cloud framework for processing of DNA and RNA sequencing data using Apache Spark; Butler (Yakneen et al (2020)) is a computational tool that eases large-scale genomic analyses on Clouds. In particular, Butler enabled processing of a 725 TB cancer genome dataset in a timely manner, with 43% increased throughput compared to prior tools. Cloud computing has been also used for executing complex Big Data mining applications. Some examples are: Agapito et al (2013) perform an association rule analysis between genome variations and clinical conditions of a large group of patients; Altomare et al (2017) propose a Cloud-based methodology to analyze data of vehicles in a wide urban scenario for discovering patterns and rules from trajectory; Kang et al (2012) present a library for scalable graph mining in the Cloud that allows to find patterns and anomalies in massive, real-world graphs; Belcastro et al (2016) propose a model for predicting flight delay according to weather conditions.

Several other works exploited Cloud computing for conducting data analysis on large amount of data gathered from social networks. Some examples are: Belcastro et al (2018) present a technique that exploits the indications contained in geotagged social media items to discover Regions-of-Interest with a high accuracy; You et al (2014) propose a social sensing data analysis framework in Clouds for smarter cities, especially to support smart mobility applications (e.g., finding crowed areas where more transportation resources need to be allocated); Belcastro et al (2019b) show the design of a cloud-based algorithm for discovering the polarization of social media users in relation to political events characterized by the rivalry of different factions; Belcastro et al (2017) present a Java library, called ParSoDA (Parallel Social Data Analytics), which can be used for developing in a easy manner social data analysis applications.

Future Directions for Research

Some of most important research trends and issues to be addressed in Big Data analysis and Cloud systems for managing and mining large-scale data repositories are:

- *In-memory analysis*. Most of the data analysis tools access data sources on disks while, differently from those, in-memory analytics access data in main memory (RAM). This approach brings many benefits in terms of query speed up and faster decisions. For example, Apache Spark stores data in RAM memory and queries it repeatedly so as to obtain better performance for many classes of applications. However, when huge amounts of data are distributed on different nodes, the communication overhead may become excessive. For this reason, high-performance hardware support and fine-grain parallel algorithms are required. For instance, *Remote Direct Memory Access* provides a direct memory access from the memory of one

computer into that of another without involving the operating system of either computer, which could permits high-throughput and low-latency networking in massively parallel computer clusters (Lin et al (2019)).

- *Scalable software architectures for fine grain in-memory data access and analysis*. Exascale processors and storage devices must be exploited with fine-grain runtime models. Software solutions for handling many cores and scalable processor-to-processor communications have to be designed to exploit exascale hardware (Talia (2019)). The design and development of Exascale systems is currently under investigation with the goal of building high-performance computers composed of a very large number of multi-core processors expected to deliver at least one exaFLOPS.
- *Programming models for data-intensive computing*. The design of data-intensive computing platforms is a very significant research challenge with the goal of building computers composed of a very large number of multi-core processors. Programming paradigms traditionally used in HPC systems (e.g., MPI[4], MapReduce) are not sufficient/appropriate for programming software designed to run on systems composed of a very large set of computing elements. To reach the Exascale size, it is required to define new programming models and languages that combine abstraction with both scalability and performance Talia et al (2019).
  From a software point of view, these new computing platforms open big issues and challenges for software tools and runtime systems that must be able to manage a high degree of parallelism and data locality. In addition, to provide efficient methods for storing, accessing and communicating data, intelligent techniques for data analysis and scalable software architectures enabling the scalable extraction of useful information and knowledge from data, are needed.
- *Massive social network analysis*. The effective analysis of social network data on a large scale requires new software tools for real-time data extraction and mining, using Cloud services and high-performance computing approaches (Martin et al (2016)). Social data streaming analysis tools represent very useful technologies to understand collective behaviors from social media data. New approaches to data exploration and model visualization are necessary taking into account the size of data and the complexity of the knowledge extracted.
- *Data quality and usability*. Big Data sets are often arranged by gathering data from several heterogeneous and often not well-known sources. This leads to a poor data quality that is a big problem for data analysts. In fact, due to the lack of a common format, inconsistent and useless data can be produced as a result of joining data from heterogeneous sources. Defining some common and widely adopted format would lead to data that are consistent with data from other sources, that means high quality data.

[4] https://www.mpi-forum.org/

References

Agapito G, Cannataro M, Guzzi PH, Marozzo F, Talia D, Trunfio P (2013) Cloud4snp: Distributed analysis of snp microarray data on the cloud. In: Proc. of the ACM Conference on Bioinformatics, Computational Biology and Biomedical Informatics 2013 (ACM BCB 2013), ACM Press, Washington, DC, USA, p 468, iSBN 978-1-4503-2434-2

Altomare A, Cesario E, Comito C, Marozzo F, Talia D (2017) Trajectory pattern mining for urban computing in the cloud. Transactions on Parallel and Distributed Systems 28(2):586–599, iSSN:1045-9219

Belcastro L, Marozzo F, Talia D, Trunfio P (2015) Programming visual and script-based big data analytics workflows on clouds. In: Big Data and High Performance Computing, Advances in Parallel Computing, vol 26, IOS Press, pp 18–31

Belcastro L, Marozzo F, Talia D, Trunfio P (2016) Using scalable data mining for predicting flight delays. ACM Transactions on Intelligent Systems and Technology (ACM TIST) To appear

Belcastro L, Marozzo F, Talia D, Trunfio P (2017) A parallel library for social media analytics. In: The 2017 International Conference on High Performance Computing & Simulation (HPCS 2017), Genoa, Italy, pp 683–690, iSBN 978-1-5386-3250-5

Belcastro L, Marozzo F, Talia D, Trunfio P (2018) G-roi: Automatic region-of-interest detection driven by geotagged social media data. ACM Transactions on Knowledge Discovery from Data 12(3):27:1–27:22

Belcastro L, Marozzo F, Talia D (2019a) Programming models and systems for big data analysis. International Journal of Parallel, Emergent and Distributed Systems 34:632–652

Belcastro L, Marozzo F, Talia D, Trunfio P (2019b) Developing a cloud-based algorithm for analyzing the polarization of social media users. In: 5th International Symposium, ALGOCLOUD 2019, Munich, Germany, to appear

Dean J, Ghemawat S (2004) Mapreduce: Simplified data processing on large clusters. In: Proceedings of the 6th Conference on Symposium on Opearting Systems Design & Implementation - Volume 6, Berkeley, USA, OSDI'04, pp 10–10

Gu Y, Grossman RL (2009) Sector and sphere: the design and implementation of a high-performance data cloud. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 367(1897):2429–2445

Hiden H, Woodman S, Watson P, Cala J (2013) Developing cloud applications using the e-science central platform. Phil Trans R Soc A 371(1983):20120,085

Kang U, Chau DH, Faloutsos C (2012) Pegasus: Mining billion-scale graphs in the cloud. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 5341–5344, DOI 10.1109/ICASSP.2012.6289127

Li A, Yang X, Kandula S, Zhang M (2010) Cloudcmp: comparing public cloud providers. In: Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, ACM, pp 1–14

Lin H, Lin Z, Diaz JM, Li M, An H, Gao GR (2019) swflow: A dataflow deep learning framework on sunway taihulight supercomputer. In: 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), IEEE, pp 2467–2475

Lordan F, Tejedor E, Ejarque J, Rafanell R, Álvarez J, Marozzo F, Lezzi D, Sirvent R, Talia D, Badia R (2014) Servicess: An interoperable programming framework for the cloud. Journal of Grid Computing 12(1):67–91

Marozzo F, Talia D, Trunfio P (2015) Js4cloud: Script-based workflow programming for scalable data analysis on cloud platforms. Concurrency and Computation: Practice and Experience 27(17):5214–5237

Marozzo F, Talia D, Trunfio P (2016) A workflow management system for scalable data mining on clouds. IEEE Transactions On Services Computing

Martin A, Brito A, Fetzer C (2016) Real-time social network graph analysis using streammine3g. In: Proceedings of the 10th ACM International Conference on Distributed and Event-based Systems, ACM, New York, NY, USA, DEBS '16, pp 322–329

Mell PM, Grance T (2011) Sp 800-145. the nist definition of cloud computing. Tech. rep., National Institute of Standards & Technology, Gaithersburg, MD, United States

Richardson L, Ruby S (2008) RESTful web services. "O'Reilly Media, Inc."

Talia D (2019) A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing 8(1):4

Talia D, Trunfio P, Marozzo F (2015) Data Analysis in the Cloud. Elsevier, iSBN 978-0-12-802881-0

Talia D, Trunfio P, Marozzo F, Belcastro L, Garcia Blas J, Del Rio D, CouvÃ©e P, Goret G, Vincent L, FernÃ ndez Pena A, Martin de Blas D, Nardi M, Pizzuti T, Spataru A, Justyna M (2019) A novel data-centric programming model for large-scale parallel systems. In: Euro-Par Workshops,

Tejedor E, Becerra Y, Alomar G, Queralt A, Badia RM, Torres J, Cortes T, Labarta J (2017) Pycompss: Parallel computational workflows in python. The International Journal of High Performance Computing Applications 31(1):66–82

Wiewiórka MS, Messina A, Pacholewska A, Maffioletti S, Gawrysiak P, Okoniewski MJ (2014) SparkSeq: fast, scalable and cloud-ready tool for the interactive genomic data analysis with nucleotide precision. Bioinformatics 30(18):2652–2653, DOI 10.1093/bioinformatics/btu343

Yakneen S, Waszak SM, Gertz M, Korbel JO (2020) Butler enables rapid cloud-based analysis of thousands of human genomes. Nature biotechnology 38(3):288–292

You L, Motta G, Sacco D, Ma T (2014) Social data analysis framework in cloud and mobility analyzer for smarter cities. In: Service Operations and Logistics, and Informatics (SOLI), 2014 IEEE International Conference on, pp 96–101