Map-Reduce for Distributed Data Processing in Big Data Analysis*

1st Rama Krishna Kamma
Computer Science and Information Systems
Texas A&M University Commerce
Commerce, Texas
rkamma2@leomal.tamuc.edu

2nd Dr Pooja Rani
Computer Science and Information Systems
Texas A&M University Commerce
Commerce, Texas
pooja.rani@tamuc.edu

***Abstract*—A Map-Reduce implementation called Hadoop creates free, open-source software for dependable, scalable, distributed computing. On Hadoop, several applications that we have suggested are used, including data extraction, transformation operations, join operations, and clustering algorithms. In this paper, we present our approach for using Map-Reduce to address the issue. They can be used to identify users who share similar interests during the pre-processing of data. The clustering methods are our main concern. At present real world scenario massive amounts of digital data are now accessible online because to the Internet's quick expansion, and its enormous storage capacity is being successfully expanded. Some computing is necessary to process, evaluate, and link a sizable volume of recorded data to produce accurate information. For example, streaming sites, music industry, e-commerce websites now frequently deal with log datasets that are up to a few terabytes in size due to the rapid development of the Internet. One issue we must deal with is how to delete chaotic data quickly and inexpensively while still obtaining important information. From preprocessing the raw data to creating the final models, there are various steps in the mining process. Map-Reduce is paired with a clustering technique called Map Reduce Service (MRS), which integrates SOM (Self-Organized Map) and fuzzy logic. Large calculation operations using MRS can be easily accommodated with the aid of a Hadoop cluster by simply adding more nodes or machines to the cluster. From the experiment that MRS is capable of processing and analyzing exceedingly big datasets at scale.**

***Index Terms*—Map Reduce, Hadoop, Big Data.**

I. Introduction

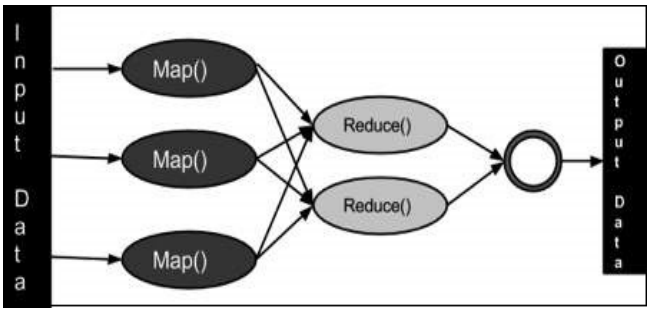
MapReduce is a programming model and software framework used for processing large amounts of data in a distributed computing environment. It breaks down a big data processing job into smaller tasks that can be distributed across multiple servers in a cluster, allowing the processing to be done in parallel.The "Map" step takes input data and converts it into a list of key-value pairs. These pairs are then passed on to the "Reduce" step, which aggregates the values for each key and produces a set of output key-value pairs.By breaking down the data processing into these two steps, MapReduce can handle massive amounts of data and distribute the processing workload across multiple machines, making it much faster and more efficient than traditional single-machine processing.MapReduce is widely used for processing large datasets in industries such as finance, healthcare, e-commerce, and social media, among others. It is the backbone of the Hadoop open-source data processing framework and has also been implemented in other big data processing systems such as Apache Spark and Google Cloud Dataflow.

Identify applicable funding agency here. If none, delete this.

II. Literature Review

The term Big Data refers to the exponential increase in the volume, speed, and diversity of data produced by numerous sources, including social media, web applications, and the internet of things. Traditional data processing techniques, which were created to operate with structured data of limited size, find it difficult to process huge data. On distributed computing systems, the MapReduce programming model is used to process massive datasets in parallel. In this paper, we are discussing about using MapReduce for big data processing and how that affects data science and analytics.

Overview of MapReduce:
Google created the MapReduce programming model to process massive datasets concurrently on distributed computing platforms. The Map phase and the Reduce phase make up this process. The input data is separated into multiple chunks during the map phase, and a map function executes in parallel on each chunk. With the incoming data, the map function generates intermediate key-value pairs. The intermediate key-value pairs are grouped by key and processed by a reduction function in parallel for each group during the Reduce phase. The final output is generated by the reduce function using the set of intermediate key-value pairs.



Source: https://www.tutorialspoint.com/

Impact of MapReduce on Big Data Processing:
MapReduce has transformed the field of big data processing

by offering a scalable and fault-tolerant method for handling enormous datasets. Data scientists and analysts can process terabytes and petabytes of data on distributed computing platforms without having to worry about the physical infrastructure. The processing of unstructured data, such as text, photos, and videos, is also made possible by MapReduce, which was not conceivable with conventional data processing techniques.

MapReduce's ability for parallel data processing, which enables quicker processing times, is one of its key features. Moreover, MapReduce offers fault-tolerance, which guarantees that even if a node in a distributed computing system malfunctions, processing will still be able to proceed without any data being lost. MapReduce furthermore offers a straightforward and user-friendly programming model that enables programmers to emphasis on the data processing algorithms rather than the supporting hardware infrastructure.

Scalability, fault tolerance, and simplicity are all goals of the MapReduce architecture. It enables developers to build applications that can handle massive volumes of data processing in parallel without having to worry about the specifics of distributed systems. The MapReduce framework can be utilized on low-cost hardware since it is built to run on massive clusters of commodity computers.

Several massive data processing activities, including data indexing, log analysis, machine learning, and natural language processing, have been carried out using MapReduce. Due to its popularity, other other MapReduce-based big data processing frameworks, including Apache Spark, Apache Flink, and Apache Beam, have been created.

Map Reduce and Its Applications

Simple data extracting and sum operation is one most usable application of map reduce. In map stage it extracts useful information from raw records and transforms the information to an output. In reduce stage it takes all values for a specific key, and generate a new list of the reduced output.

When processing datasets, we frequently need to join two files together, just like in the relational database query language SQL. Join operations frequently use up a lot of computational resources. Massive amounts of data will take up a lot of time and space if we process it on a single system. Join operations can occasionally fail. However, this issue can be resolved using the Map-Reduce technology.

III. Ease of Use

A. Maintaining the Integrity of the Specifications

The IEEEtran class file is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

IV. Prepare Your Paper Before Styling

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections IV-A–IV-E below for more information on proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not number text heads—LaTeX will do that for you.

A. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

B. Units

- Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as "3.5-inch disk drive".
- Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.
- Do not mix complete spellings and abbreviations of units: "Wb/m^2" or "webers per square meter", not "webers/m^2". Spell out units when they appear in text: ". . . a few henries", not ". . . a few H".
- Use a zero before decimal points: "0.25", not ".25". Use "cm^3", not "cc".)

C. Equations

Number equations consecutively. To make your equations more compact, you may use the solidus (/), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in:

$$a + b = \gamma \tag{1}$$

Be sure that the symbols in your equation have been defined before or immediately following the equation. Use "(1)", not "Eq. (1)" or "equation (1)", except at the beginning of a sentence: "Equation (1) is . . ."

D. LaTeX-Specific Advice

Please use "soft" (e.g., `\eqref{Eq}`) cross references instead of "hard" references (e.g., `(1)`). That will make it possible to combine sections, add equations, or change the order of figures or citations without having to go through the file line by line.

Please don't use the {eqnarray} equation environment. Use {align} or {IEEEeqnarray} instead. The {eqnarray} environment leaves unsightly spaces around relation symbols.

Please note that the {subequations} environment in LaTeX will increment the main equation counter even when there are no equation numbers displayed. If you forget that, you might write an article in which the equation numbers skip from (17) to (20), causing the copy editors to wonder if you've discovered a new method of counting.

BibTeX does not work by magic. It doesn't get the bibliographic data from thin air but from .bib files. If you use BibTeX to produce a bibliography you must send the .bib files.

LaTeX can't read your mind. If you assign the same label to a subsubsection and a table, you might find that Table I has been cross referenced as Table IV-B3.

LaTeX does not have precognitive abilities. If you put a \label command before the command that updates the counter it's supposed to be using, the label will pick up the last counter to be cross referenced instead. In particular, a \label command should not go before the caption of a figure or a table.

Do not use \nonumber inside the {array} environment. It will not stop equation numbers inside {array} (there won't be any anyway) and it might stop a wanted equation number in the surrounding equation.

E. Some Common Mistakes

- The word "data" is plural, not singular.
- The subscript for the permeability of vacuum μ_0, and other common scientific constants, is zero with subscript formatting, not a lowercase letter "o".
- In American English, commas, semicolons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited, such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)
- A graph within a graph is an "inset", not an "insert". The word alternatively is preferred to the word "alternately" (unless you really mean something that alternates).
- Do not use the word "essentially" to mean "approximately" or "effectively".
- In your paper title, if the words "that uses" can accurately replace the word "using", capitalize the "u"; if not, keep using lower-cased.
- Be aware of the different meanings of the homophones "affect" and "effect", "complement" and "compliment", "discreet" and "discrete", "principal" and "principle".
- Do not confuse "imply" and "infer".
- The prefix "non" is not a word; it should be joined to the word it modifies, usually without a hyphen.
- There is no period after the "et" in the Latin abbreviation "et al.".
- The abbreviation "i.e." means "that is", and the abbreviation "e.g." means "for example".

An excellent style manual for science writers is [7].

F. Authors and Affiliations

The class file is designed for, but not limited to, six authors. A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

G. Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is "Heading 5". Use "figure caption" for your Figure captions, and "table head" for your table title. Run-in heads, such as "Abstract", will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced.

H. Figures and Tables

a) Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation "Fig. 1", even at the beginning of a sentence.

TABLE I
TABLE TYPE STYLES

Table Head	**Table Column Head**		
	Table column subhead	***Subhead***	***Subhead***
copy	More table copy[a]		

[a]Sample of a Table footnote.

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when

Fig. 1. Example of a figure caption.

writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks . . .”. Instead, try “R. B. G. thanks. . .”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

Please number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first . . .”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

REFERENCES

[1] Mingyue Luo,Gang Liu, “Distributed log information Processing with Map-Reduce,” 2010 IEEE International Conference on Information Theory and Information Security DOI: 10.1109/ICITIS17077.2010, 17-19 Dec. 2010.

[2] J Dean, S Ghemawat,“ MapReduce: Simplified data processing on large clusters, ” Communications of the ACM, 2008.

[3] CK Fong,“A Study in Deploying Self-Organized Map(SOM) in an Open Source J2EE Cluster and Caching System, ” 2007 IEEE/ICME International Conference on Complex Medical Engineering, pp.778-781,2007.

[4] https://en.wikipedia.org/wiki/MapReduce

[5] Tianyang Sun, Chengchun Shu,“ An Efficient Hierarchical Clustering Method for Large Datasets with Map-Reduce, ” Parallel and Distributed Computing, Applications and Technologies, 2009 International Conference on, pp.494-495.

[6] https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

[7] S Papadimitriou, “DisCo: Distributed Co-clustering with MapReduce. A Case Study Towards Petabyte-Scale End-to-End Mining, ” Data Mining ,ICDM ’08. Eighth IEEE International Conference on, pp. 512 - 521 ,2008.

[8] Petri Vuorimaa, “Fuzzy self-organizing map, “Fuzzy Sets and Systems,” pp 223-231, 1994.

[9] Gul Shaira Banu Jahangeer, T. Diliphan Rajkumar, “An Implementation of Map Reduce on the Hadoop for Analyzing Big Data,” International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-4S2, December 2019.

[10] Cheng-Tao Chu, Sang Kyun Kim and Vi-An Lin, “Map-Reduce for Machine Learning on Multicore,” NIPS, pp. 281-288, 2006.