JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-6, Issue-7, July 2018 E-ISSN: 2347-2693

Plagiarism Detection on BigData Using Modified Map-Reduced Based N-Tuple Algorithm

Thanu Kurian[1*], Manukuru Hymavathi[2], Tina Thomas[3]

[1,2,3]Dept. Of Computer Science and Engineering, MVJ College of Engineering, VTU , Bangalore,India

[*]Corresponding Author: thanukurian@gmail.com, Tel.: 7259633377

Available online at: www.ijcseonline.org

Accepted: 23/Jul/2018, Published: 31/July/2018

Abstract**—** Plagiarism or the expropriation of another author's data and the presentation of it as one's own, is a serious violation of ethics of professionalism. Attempting to take other person's works, without proper citation is considered as one way of Plagiarism. With the rapid use of internet access and large amounts of big data, copying of content partially or fully has become a common practice. The proposed technique, map-reduced N-Tuple algorithm for distributed computing platform compares the number of attributes of the comparing tuples, at first. If the number of attributes is different, we are sure that the tuples cannot be equal. If the number of attributes is the same, we further sort the values inside of each tuple of both relations. This sorting is necessarily to make sure, that afterwards we can use the equal functionality, provided by Standard Library to find out, whether all corresponding pairs of two tuples compare equal. Here different capacity data sets are tested for plagiarism, which gives output within short time and more accuracy compared to the Standard Copy Analysis Mechanism. Our proposed algorithm is used to compare documents for processing big data using Hadoop and detect plagiarism for performance enhancement.

Keywords**—**Plagiarism, N-Tuple, Big data, Hadoop, MapReduce

I. INTRODUCTION

Plagiarism, defined as "Stealing or attempting to steal someone else's intellectual property and claiming it to be one's own" has become a serious and common issue nowadays as more and more digital data is available online[1]. The meaning of the word "plagiarism" is even wider which includes rearranging the words in a sentence without changing the meaning, replacing a few words by its synonyms, bringing together bits of work by different authors without proper reference. As this is considered as one of the cyber-crimes, there are many plagiarism detection softwares which are available in the market to avoid plagiarised contents[2]. As long as you give credit to the previous work when you are mentioning it, the content is not grieved to be plagiarised[3].There should be a clear boundary between the existing knowledge and starting of your work. There are many types of plagiarism as shown in Figure 1 which are serious violations of academic honesty.

Direct plagiarism is the word-for-word transcription of a section of someone else work, without attribution and without quotation marks. The deliberate plagiarism of someone else work is unethical, academically dishonest, and grounds for disciplinary actions, including expulsion.

Self-plagiarism occurs when a student submits his or her own previous work, or mixes parts of previous works, without permission from all professors involved.

Mosaic Plagiarism occurs when a student borrows phrases from a source without using quotation marks, or finds synonyms for the author's language while keeping to the same general structure and meaning of the original. Sometimes called "patch writing," this kind of paraphrasing, whether intentional or not, is academically dishonest and punishable – even if you footnote your source!

Accidental plagiarism occurs when a person neglects to cite their sources, or misquotes their sources, or unintentionally paraphrases a source by using similar words, groups of words, and/or sentence structure without attribution.

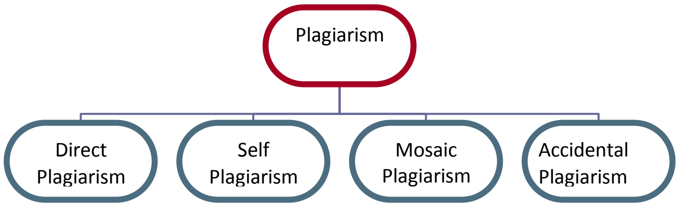

Figure1. Classification of Plagiarism

Big data management is the organization and manipulation of huge volumes of structured data, semi-structured data and unstructured data. The aim of big data management is to make sure the quality of high level data and availability of data for business intelligence and big data analytics applications. Data scientists almost always describe "big data" as having at least three distinct dimensions: volume, velocity, and variety. Some then go on to add more Vs to the list, to also include—in my case—variability and value.

Volume: Big data first and foremost has to be "big," and size in this case is measured as volume.

Velocity: Velocity in the context of big data refers to two related concepts familiar to anyone: the rapidly increasing speed at which new data is being created by technological advances, and the corresponding need for that data to be digested and analyzed in near real-time.

Variety: With increasing volume and velocity comes increasing variety. This third "V" describes just what you'd think: the huge diversity of data types that organizations see every day.

Variability: The way care is provided to any given data depends on all kinds of factors—and the way the care is delivered and more importantly the way the data is captured may vary from time to time or place to place.

Value: Last but not least, big data must have value. That is, if you're going to invest in the infrastructure required to collect and interpret data on a system-wide scale, it's important to ensure that the insights that are generated are based on accurate data and lead to measurable improvements at the end of the day.

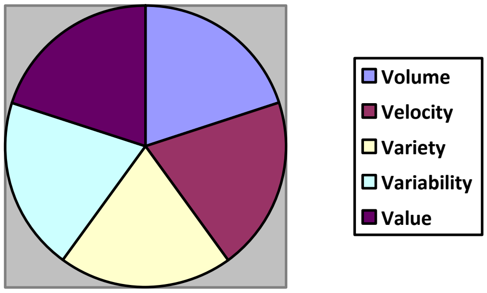

Figure 2. Dimensions of Big Data

Data Analysis

a) **Hadoop**:This is an open source platform (tool) for treating big data and its analytics. It is user friendly and flexible to work with different data sources, either gathering various sources of data or accessing the data from a database in order to run processor-intensive machine learning process. This tool has different types of applications such as location based data from weather, traffic sensors and social media data.

b) **Map Reduce**: This is the programming environment that permits larger jobs implementation scalability against group of server. Map Reduce implementation has two main tasks: The Map task converts input dataset is into a different set of value pairs. The Reduce task combines several outputs of the Map task to form reduced tuples.

c) **Hive**: Hive is the SQL-like bridges that permit predictable business applications to run SQL queries against a Hadoop cluster.

There are many other tools like PIG, WibiData, Platfora, Rapidminer.

The idea of the following algorithm is to compare the number of attributes of the comparing tuples, at first. If the number of attributes is different, we are sure that the tuples cannot be equal. If the number of attributes is the same, we further sort the values inside of each tuple of both relations . This sorting is necessarily to make sure, that afterwards we can use the equal functionality, provided by STL to find out, whether all corresponding pairs of two tuples compare equal. Finally, the general function tell us, if the two relations have tuples in common or not.

Rest of the paper is organized as follows, Section I contains the introduction of Plagiarism, BigData and Data Analysis, Section II contain the related work of techniques used for checking plagiarism, Section III explains the methodology with N-Tuple algorithm used to check plagiarism in BigData, Section IV describes results and discussion with graph and a table, Section V concludes research work with future directions.

II. RELATED WORK

In this section we will discuss the different software used in textual plagiarism. Earlier, plagiarism detection was done manually without the help of any tools or software[4].This worked fine for small number of documents, but as the count of documents increases, manual detection becomes more time consuming and less accurate

In plagiarism detection there are two major categories for algorithms to detect the plagiarized data; algorithms that analyse grammar structure of the text and algorithms that analyse the text fingerprints.The software tool, Free Text Plagiarism Detection Software (FTPDS) uses documents' fingerprints to detect the likelihood that the documents are plagiarized from each other [5]. The system is able to detect plagiarism between two given documents, given document and group of local documents, and between given document and online available documents. But the number of checked documents in online search is limited by the speed of the connection and available memory. And there is no suitable method for online search and comparison without downloading the documents to the local disks found.

Plagiarism ,which is using someone else work without crediting the original author, has a number of negative effects on education. It limits one's critical thinking and thought process in developing his/her ideas, thus negatively impacting the educational experience of the student in the college/university. Therefore it is vital to detect cases of plagiarism and apply appropriate punishments in order to deter students. There are two techniques for plagiarism detection and prevention, the first method is based on the allocation of a unique assignment for each student, while the second approach is based on the use of individual presentation of coursework findings .It faces issues that it is less effective in detection of different forms of plagiarism such as ideas-theft or collusion [6].

The most common plagiarism in written text document is formed by copying some or all parts of the original document, sometimes with some modifications. Identification of documents which were copied is stressful and time-consuming process to humans due to the large number of documents which have to be analyzed. The documents in digital format make the process of plagiarism quite simple, it means that such cases of plagiarism can be traced automatically. The development of the intelligent system for searching for plagiarism by combining two algorithms of searching fuzzy duplicate is considered in [7]. The issue with Fuzzy Duplicates Algorithm is it is suitable only for a small sized document.

An efficient plagiarism detection tool, CPLAG, for C programming language codes. The tool assesses the structure of the C programs based on a set of attributes and performs a binary encoding of the C code statements. But the efficiency and the quality of plagiarism reduces for larger datasets or bigger documents [8].

SCAM , plagiarism detection algorithm for big-data which calculates relative measure to detect overlap by making comparison on asset of words that are common between test document and registered document. SCAM algorithm gives less accurate data and takes longer time to give results, compared to our proposed algorithm.[9][10]

III. Methodology

We propose an N-Tuple algorithm for bigger data sets, and gives output in short time with speed and accuracy compared to SCAM algorithm.The algorithm takes in 4 arguments.

1. File name for a list of synonyms
2. input file1
3. input file2(repository file)
4. (optional) the number N, the tuple size. If not supplied, the default should be N=3.

The synonym file has group of synonyms in each line. For example a line in the synonym file can contain "have own possess" means these words should be treated as equal.

The input file1 can be said to be plagiarised based on the number of N-tuples in file1 that appear in file2 (in other words we are looking for tuples in file1 that matches ones in repository (file2)) against which we are checking the percentage of plagiarism in file1), where the tuples are compared by accounting for synonyms as described above. For example, the text "I have a book" (file1) has two 3-tuples, ["I have a", "have a book"] both of which appear in the text "I have a book" (file2).

The output of the program should be the percent of tuples in file1 which appear in file2. So for the above example, the output would be saying "100%". If we consider another example, for texts "I have a book" and "I gave a book" and N=3 we would output "0%" because both tuple in the first text does not appear in the second one.

Step1:Implement the N-Tuple algorithm for detecting plagiarism

Algorithm: N-Tuple_Bigdata

```
Step1(a): matched_tuple_count=0,unmatched_tuple_count=0
      Read file1(input file) and file2(repository file).
             If( file1==null||file2==NULL)
                        return;
Step1(b): for(i=0;i<=file2.wordcount()-tuplesize;i++)
                   read_next_tuple();
               if(TupleMap.contains(tuple)
                 tuplemap.put(tuple,prev_value+1)
               else
                  tupleMap.put(tuple,1)
step1(c): for each word in file1
        construct a sentence of tuplesize starting from word
           if(tupleMap.containsKey(sentence))
                    matched_tuple_count++;
           else
                    unmatchedtuplecount++;
matching_ratio=matched_tuple_count/(matched_tuple_count
               +unmatched_tuple_count);
percentage=matching_ratio*100;
                  return percentage;
```

Step2: Setup and configure hortonworks data platform software for bringing the Hadoop distributed environment up and running.

Step3: Design and implement the web server module for enabling end users upload the files to the repository.

Step4: Integrate all the previous modules into one fully functional system.

IV Results and Discussion

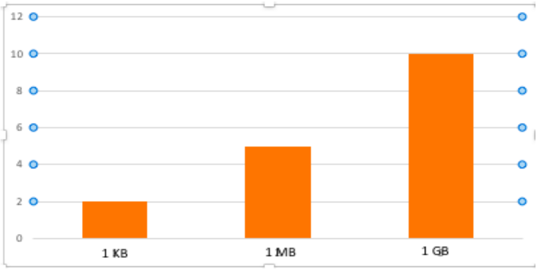

Figure 3. Comparison of execution time for N Tuple algorithm.

The comparison results of execution for N Tuple algorithm attempted with variable data set is presented in figure 3.Table 1 gives a brief description comparing the execution time of SCAM and N-Tuple algorithm and are scaled separately for variable data set.

In SCAM, plagiarism detection takes place by calculating relative measure to detect overlap by making comparison on asset of words that are common between test document and registered document, whereas in N-tuple algorithm the input file can be said to be plagiarised based on the number of N-tuples in input file that matches ones in repository file against which we are checking the percentage of plagiarism in input file, where the tuples are compared by accounting for synonyms. As a result, SCAM algorithm gives less accurate data and takes longer time to give results, compared to our proposed algorithm.

Table 1:Comparision between N Tuple and SCAM algorithm.

Data size	**1KB**	**1MB**	**500MB**	**1GB**
SCAM (Execution time in sec)	20 sec	100-200 sec	400-500 sec	900-1000 sec
N-Tuple (Execution time in sec)	1-2 sec	30-40 sec	60-70 sec	100-200 sec

V. CONCLUSION AND FUTURE SCOPE

In this work N-Tuple algorithm is modified for distributed computing platform using Hadoop [11]. In this work different capacity datasets are tested for plagiarism using modified N-Tuple on Hadoop. It is found that execution time doesn’t increase considerable for bigger dataset also and data will be distributed across the cluster of machines. This technique takes sometimes for finding results gives output in short time with speed and accuracy and we are easily process and handle big data sets. Hadoop is used for performance enhancement.

REFERENCES

[1] Manuel Zini,Marco Fabbri,Massimo Moneglia, Alessandro Panunzi,”*PlagiarismDetectionThroughMultilevelTextComparison*”, Universit`a di Firenze, Italian Department

[2] Mr. Dnyaneshwar R. Bhalerao and Prof. S.S.Sonawane,”*A Survey ofPlagiarism Detection Strategies and Methodologies in Text Document*”, Department of Computer Engineering, PICT, Pune-411043.

[3] Asim M. El Tahir Ali, Hussam M. Dahwa Abdulla, and V´aclav Sn´aˇsel,“*Overview and Comparison of Plagiarism Detection Tools*”, Department of Computer Science, VˇSB-Technical University of Ostrava.

[4] Si, Antonio, Hong Va Leong, and Rynson WH Lau. "*Check: a document plagiarism detection system*." Proceedings of the 1997 ACM symposium on applied computing. ACM, 1997.

[5] Mohamed Elkhidir, Mohannad M. Ibrahim, Tarig A. Khalid, Shawgi Ibrahim, Mohamed Awadalla, “*Plagiarism Detection using Free-Text Fingerprint Analysis*” Department of Electrical & Electronic Engineering University of Khartoum Khartoum, Sudan

[6] Basel Halak and Mohammed El-Hajjar, “*Plagiarism Detection and Prevention Techniques In Engineering Education*” Electronics and Computer Science University of Southampton, Southampton, UK

[7] Natalya Shakhovska, Iryna Shvorob, “*The method for detecting plagiarism in a collection of documents*” COMPUTER SCIENCE & INFORMATION TECHNOLOGIES, (CSIT’2015) LVIV, UKRAINE

[8] Shikha Jain, Parmeet Kaur, Mukta Goyal, Dhanalekshmi G., “*CPLAG: Efficient Plagiarism Detection using Bitwise Operations*” Department of Computer Science and Information Technology, Jaypee Institute of Information Technology, Noida, India.

[9] Jayshree Dwivedi, Prof. Abhigyan Tiwary,,“*Plagiarism Detection on Bigdata Using Modified Map-Reduced Based SCAM Algorithm*”, Department of Computer Science and Engineering, SIRTS Group of Institute Bhopal, India.

[10] Mrs. Parminder Kaur ., *“Methods for Web-Spam Detection on web: Principles and Algorithms”,* International Journal of Scientific Research in Computer Science and Engineering, Vol.6, Issue.2, pp.119-125, 2018.

[11] Amit Palve, Ajit Patil, Amol Potgantwar, "*Big Data Analysis Using Distributed Approach on Weather Forecasting Data*", International Journal of Scientific Research in Network Security and Communication, Vol.5, Issue.3, pp.39-43, 2017.

Authors Profile

Mrs Thanu Kurian pursed Bachelor of Technology from University College of Engineering, Kerala in 2002 and Master of Technology in Computer Network Engineering from CMR Institute of Technology, Bangalore in year 2012. She is currently working as Assistant Professor in Department of Computer Science and Engineering, MVJ College of Engineering, Bangalore. Her main research work focuses on Wireless Networks, Network Security, Big Data Analytics. She has 6 years of teaching experience

Mrs M.Hymavathi pursed Bachelor of Technology in Computer Science and Engineering from Kuppam Engineering College, JNTU University in 2005,and Master of Technology in Digital Communication and Networking from SJCIT, VTU University, Bangalore in 2011. She is currently working as Assistant Professor in Department of Computer Science and Engineering, MVJ College of Engineering, VTU University,Bangalore since 2013.She is an Oracle Certified Associate.She published paper related to Cryptography and Networking. Her main research work focuses on Cryptography Algorithms, Network Security, Security and Privacy,Big Data Analytics.She has 7 years of teaching experience .

Ms Tina Thomas pursed Bachelor of Engineering in Computer Science and Engineering from MVJ College of Engineering, VTU, Bangalore in 2018.Her main research work focuses on Big Data Analytics.