International Journal of Computer Sciences and Engineering
Vol.**11**, Issue **4**, pp.**61-62**, April **2023**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

ISSN: 2347-2693 (E)

Case Study

Plagiarism Checker Data Indexing Technology for Indian Regional Language

Prashanth Kumar H.M.[1*], Subramanya Bhat S.[2]

[1,2]College of Computer Science, Srinivas University, Mangalore, India

**Corresponding Author: prashanth.hm02@gmail.com*

Received: 28/Feb/2023; **Accepted**: 08/Apr/2023; **Published**: 30/Apr/2023. **DOI**: https://doi.org/10.26438/ijcse/v11i4.6162

Abstract: Plagiarism is considered a serious academic and ethical offense, as it undermines the values of originality, honesty, and integrity in academic and creative work. India has a diverse linguistic landscape, with over 22 official languages and many more regional languages spoken across the country. Several Indian states have taken steps to promote regional language education in recent years. In this case study we are exposing a very accurate plagiarism checker for all indian regional languages. We are facing many challenges to develop this sort of software. So, mainly the data indexing methods are very interesting in this case. Here we are exposing how data indexing methodology works using 'Taylor series' formula in cloud-based storage for Indian regional languages.

Keywords: Indexing, Encryption, Data Sequence, Search Key.

1. INTRODUCTION

Data indexing technology is the process of creating an index or catalog of the content of a collected of texts or other, to facilitate efficient searching, retrieval, and analysis. This is the process of creating an index or database of searchable terms or keywords that can be used to quickly find specific pieces of text. There are several ways to do text indexing, but one common method is using search engines or specialized software tools. In the market they have indexing methodologies like Apache Lucene, Elasticsearch, and Solar etc.., but they have some character limitations. Ex: Languages are Urdu, Arabic or Persian are Right to Left languages, you can't make sure others are providing best accuracy in this, so avoiding such sort of issue we can make it our own indexing protocol. Eventually, our indexing process involves identifying the key concepts, terms, or entities that are important for describing the content of the texts, and creating an index that URL maps these concepts to the locations in the texts where they occur. This allows users to search for specific words or phrases and retrieve relevant texts quickly, rather than having to read through the entire collection. Text regional language text indexing is a fundamental technique in many information retrieval systems of plagiarism checking. Regional language characters contain core characters and supporting character, in each language containing more than 125 characters and referenced by minimum four length of Unicode characters, Example in Kannada letter 'ಅ' Unicode is U+0C85. Here, if more than 22 languages come with all characters, the indexing level in a single cloud storage platform goes to a very high level, if huge numbers of indexing level will kill the searching operation during the plagiarism checker process. And it will take more operational expenses and delay. On the other end we have to find and avoid non-indexing characters in regional languages. These non-indexing characters will create exception status and sometimes it will create crashes in operation during implementation. Commonly non-indexing refers to ': ? > < * \ /' but in the case of regional language we can cover almost 22*150 characters should be a level of initial indexing methods. The indexing will be done by organizing a tree structure of data which finally refers to the position of a cloud-based file path. A single text file is divided into multiple selected sequences according to our algorithm, and sequence removed duplication to avoid indexing ambiguity. This sequence is passed to the index number of levels until destination. Finally, the indexed file will store the cloud data path to refer to the same file path for referencing and the indexed results must return to be accurate and relevant. Our code executes approximately 20 million sequences in 100-120 seconds, which means fast and indexing successive rate is 99.99%. we reached the best time and space complexity as our indexing expectation. The best indexing process we have done using the Taylor series is the polynomial formula explanation.

2. EXPERIMENTAL METHOD

Index Encryption: A single file data can be split into several sequences. Ex: if you have 100 kb of text data may split into

more than 3000 sequences, each sequence is a combination of consecutive characters. In regional Kannada language letter 'ಅ' is finding his Unicode character is 'U+0C85', if you are running 64 bit of your cloud server then you can divide Unicode '0C85' to 16 bits binary level. Means 0 (decimal 00) => 0000000000000000, C(decimal 83)=> 0000000001010011, 8(decimal 72)=> 0000000001001000 and 5(decimal 69) => 0000000001000101. After combining all binary sequences, we get 64 digits (same as 64 bit of your cloud server data transfer bus) of value. Finally, u will get mod%4 operation using all 64 binary digits. We will get binary sequences 0000000000010000, 0000001000100011, 0000000100000000 and 000000110000000. Later if you convert decimal from signed 2's complement with mod%10 operation you will get values 0C85(input) to 6769(output). We can conclude her search key is 0C85, indexed data is 6769, and sequence is 16 bits binary values.

3. RESULTS AND DISCUSSION

According to our algorithm process, the Taylor series is the polynomial, and it is a function of an infinite sum of sequence terms. Each successive sequence term will have a more exponent sequencing degree than the indexing term. A data reference is always pointing to a search key variable, then the sequence is pointing to a data reference variable similarly it's a triangle workflow of all referencing paths. When the formula enters an iterative process the Taylor series algorithm creates an index of the text data using encryption method, which includes the key terms and keywords they have automatically identified. This typically involves creating an inverted index, which maps terms to the search key where they appear. This process works like A=>B, B=>C and C=>A. In all search key, data reference and sequences workflow under Taylor series flow, here f(x) assigning all set of differentiable function [f(x) = f(a) + f'(a) (x − a) + [f''(a)/2! (x − a)2] +…+], where function (f called search key), neighborhood number (a called data reference) and composite value (x called sequence). Finally, the f(x) search method holds all three operations for finding searchable value within a short time during the plagiarism checking process.

4. CONCLUSION

From the above case study information finally, we conclude that our regional language indexing methodology is one of best data indexing technology in our plagiarism checking process. Large number of data sequences will be indexed with less duration compared to other open-source indexing technology. This sort of complex work done by Taylor series logic and security part used binary encryption sequence, both mentioned in our solution index encryption part.

REFERENCE

[1] J. Li, Z. Xu, Y. Jiang, and R. Zhang, "*The overview of big data storage and management. cognitive informatics cognitive computing*" in IEEE 13th International Conference on, (pp. 510-513, 2014.

[2] C. Liu, Zhang, C. Yang, D. Georgakopoulos, and J. Chen, "*Public auditing for big data storage indexing -," in A Survey. Computational Science and Engineering (CSE),* 2013 IEEE 17th International Conference on, (pp. 1121-1135)., 2015, Dec.

[3] H. Tan, W. Luo, and L. M. Ni, "*Taylor series and its functions in data analytics*.," in Proceedings of the22nd ACM International Conference on Information and Knowledge Management (pp. 2149-2183). New York, NY, USA: ACM., 2012.

[4] W. Zhou, C. Yuan, R. Gu, and Y. Huang, "*Large scale nearest neighbors search based on neighborhood graph*," in Advanced Cloud and Big Data (CBD), 2013 International Conference on, pp. 181–186, Dec 2013.

[5] H. Nakada, H. Ogawa, and T. Kudoh, "*Stream processing with bigdata: Sss-mapreduce*," in Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on, pp. 618–621, Dec 2012.

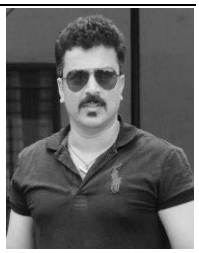
[6] T. Chardonnens, "*Big data analytics on high velocity streams*," Master's thesis, University of Fribourg (Switzerland), June 2013.

[7] F. Amato, A. De Santo, F. Gargiulo, V. Moscato, F. Persia, A. Picariello, and S. Poccia, "*Semtree: An index for supporting semantic retrieval of documents*," in Data Engineering Workshops (ICDEW), 2015 31st IEEE International Conference on, pp. 62–67, April 2015.

[8] Cambazoglu BB, Kayaaslan E, Jonassen S, Aykanat C (2013*) "A term-based inverted index partitioning model for efficient distributed query processing".* ACM Trans Web 7(3):1–23. doi:10.1145/2516633.

[9] Bast H, CelikikM(2013) *"Efficient fuzzy search in large text collections".* ACM Trans Inf Syst 31(2):1–59. doi:10.1145/2457465.2457470

AUTHORS PROFILE



Mr. Prashanth Kumar HM: studied his B.E. from KVG College of Engineering, Sullia and M. Tech., from SJCE Mysore. Now pursuing Ph.D. in Engineering and Technology from Srinivas University, Mangalore, India. He is currently working as Founder and Chief Technology Officer in DrillBit SoftTech India Pvt., Ltd., Bangalore since 2017. He has published research papers in reputable international journals and it's also available online. His main research work focuses on Plagiarism related work like Crawling Technology, Data Indexing, Big Data Analytics, Data Mining, IoT and AI based educational technologies.

Dr. Subramanya Bhat: did his MSc (CS) from Mangalore University during 2002, and M.Phil (CS) from Alagappa University, Madurai during 2008. Presently he has completed his PhD work and submitted the thesis to Rayalaseema University, Kurnol, AP. He is working at MCA department as Associate Professor and serving as Dean of Institute of Computer Science and Information Sciences at Srinivas University, Mangalore, Karnataka. During his service, he has done about 28 publications in reputed National and International Journals, attended more than 100 conferences. He is the chairman of BOS - Computer Science and Information Sciences at Srinivas University, Member of BOS at Vivekananda College Puttur (Autonomous) for BCA program. He is also a member of BOE for PG studies at Mangalore University.

