ISSN: 2581-3404 (Online) **IF: 5.445 (SJIF)**

International Journal of Innovative Research in Technology and Management, Volume-6, Issue-4, 2022.

A Structure for Detecting Document Plagiarism Using the Rabin Karp Algorithm

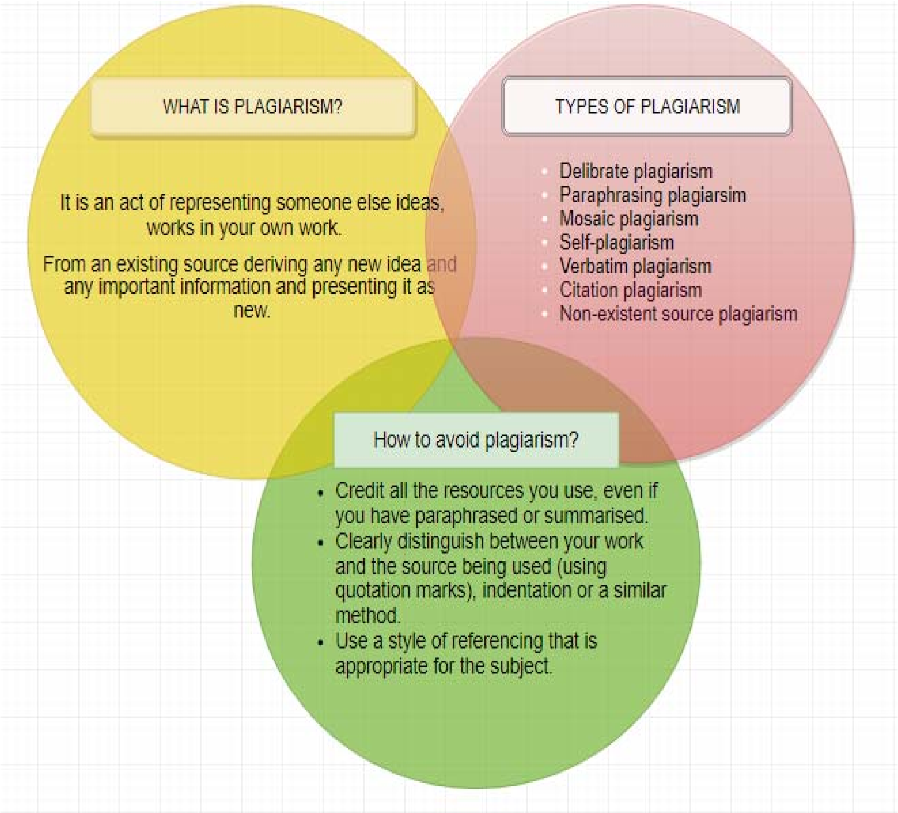
Atul Kumar Chaudhary[1], Chetan Agrawal[2], Pooja Meena[3]
Dept. of CSE, Radharaman Institute of Technology & Science, Bhopal, India[1, 2, 3]
atulkumarchaudhary63@gmail.com[1], chetan.agrawal12@gmail.com[2], meena.pooja1@gmail.com[3]

Abstract: *Replicating the thoughts or words of another author's work, as well as copying and pasting research that has been circulated is considered to be plagiarism. Some frequent kinds of plagiarism that may be identified include cheating, failing to attribute sources, and patchwork. Now, plagiarism is transferring from mediaeval history to this current era (in the Research field, academic culture, workplace, Students locations, etc.). With the expanding usage of the internet, numerous online articles are accessible boundlessly. Plagiarism is shifting from the mediaeval period to this current era. However, it is possible to go around this by using a technique that detects plagiarism in order to get a paper that is not plagiarized. Word matching, string matching, semantic-based and knowledge-based text classification are some examples of the many algorithms that have been developed for such document processing in order to detect similarities. These algorithms all operate on the same basic premise. Previous methods have their limitations when it comes to dealing with data analysis activity that only uses one level of processing. Compressive sensing based Rabin Karp (CS-RKP), a sophisticated and unique method is recommended for use in the system that is now under consideration. This approach used a sampling module for the purpose of decreasing the size of the dataset, as well as an additional cost function for the identification of document redundancy. Additionally, it calculated both syntax and semantic for the purpose of locating similarities throughout the text. The efficiency of the suggested approach is shown by the evaluation of computation time and similarity measure with respect to n-gram for varying values of N in the result observation.*

Keywords: Plagiarism, N Gram, Semantic Analysis, Document Sampling, Data Redundancy, Compressive Sensing.

Introduction

Plagiarism is an illegal act of copy infringement that means copying or using some other author's published data in your work. Some persons who replace or delete some words or modify it adding some more information in form of words or phrases in a given sentence is known as plagiarist [1]. Plagiarism is copying someone else facts, figures, words, graphs, and tables without giving any credit or simply changing some words or phrases in your language or trying to alter the grammatical styles written by some other author or mixing two or more patterns of writing into a single document [2]. For example, in a school students are solving the assignment together by copying another student’s copy. Different cases of plagiarism can be seen by following changes: 1) Lexical changes 2) Syntactic changes 3) Semantic changes. In lexical changes suspected author can directly copy some part of the document or modify it adding some more words to the original document. In Syntactic changes, the plagiarist either changes the grammar style or reforms sentences string-matching approach is used to detect these changes. Whereas, Semantic change occurs by reordering the sentences or words or phrases in the sentences or replacing the whole paragraph with another by using the concept of ‘synsets’ means using synonyms of the given terms [3]. Hence, plagiarism is an unfair practice and it is critically important to curb this problem. Generally, Plagiarism is viewed as scholastic dishonesty and a break of journalistic morals. It is liable to sanctions like punishments, suspension, and even removal. Recently, instances of plagiarism as immoral and originality have been recognized in the scholarly world. It is a big and difficult problem and it is much needed to mitigate as it violates the rules of many academic institutions and research publications. There are three levels of plagiarism: a) Near copy b) Light revision c) Heavy revision. These types of plagiarism cases fall under the category of “Literal Plagiarism” and it is easily suspectable as the author does not need any special knowledge to detect it. Another type of plagiarism is “Intelligent plagiarism” which is quite ferocious for researchers, students, and academic scholars and it is very difficult to detect it. Further deeply and collectively study about the different types of plagiarism ranges from low to high level but all are responsible to play with academic honesty [4].



WHAT IS PLAGIARISM?

It is an act of representing someone else ideas, works in your own work.

From an existing source deriving any new idea and any important information and presenting it as new.

TYPES OF PLAGIARISM

- Delibrate plagiarism
- Paraphrasing plagiarsim
- Mosaic plagiarism
- Self-plagiarism
- Verbatim plagiarism
- Citation plagiarism
- Non-existent source plagiarism

How to avoid plagiarism?

- Credit all the resources you use, even if you have paraphrased or summarised.
- Clearly distinguish between your work and the source being used (using quotation marks), indentation or a similar method.
- Use a style of referencing that is appropriate for the subject.

Figure 1: Concept of plagiarism.

II. Related Work

In the given paper [5], the author used the Rabin-Karp algorithm and find the hash values of the documents to get them between them. Hashing technique is the best searching technique that can be used to find the hash value of the given string or find the number of a given string. First of all the data is preprocessed tokenization of data takes place, stemming words are removed as well as stop words are removed. Then, after this hash function is used to find values of different words and use large prime numbers to find the value by considering the N-gram factor into consideration. For example, if the word is 'HAPPY' so K=5. Then the hash value is calculated by adding the individual value of their ASCII value multiplied by a predefined prime number. Two documents are matched by their matching hashing values and then, at last, find the percentage of plagiarism. It is not just a good algorithm but also fasts to find similarities.

In the given paper [6], the author analyzed two different algorithms- The winnowing algorithm and the Karp Rabin algorithm for detecting the similarity between the two documents. First and foremost, the two calculations play out the preprocessing ventures by separating the information and eliminating stop words, accentuations mark, stemming words, and more successive words. After this n-gram boundary is picked where n is a sub-grouping of letters, words, and a blend of a few words in a given string. On considering the n-gram boundary they determined the hash esteem utilizing hashing strategy. In the Rabin-Karp calculation, the hash esteems will be utilized to make the finger impression as indicated by which both the information record and dubious archive are coordinated. The moving hash technique is utilized to figure the hash an incentive for every one of the upsides of the k-gram given in the source text. However, Winnowing calculation creates the solitary unique mark as per the base hash esteems which implies it chooses a hash esteem from the window.

In the given paper [7], the author introduces an EMAS framework for plagiarism detection means textual plagiarism in which they introduced layered architecture and all the layers perform different tasks in which layer 1 performs preprocessing of data and data cleaning, layer 2 performs stop word removal, redundancy and stemming after this at layer 3 Evolutionary Multi-agent system paraphrasing matching is performed, parsing is done and linear string matching is done in this layer and citation-based analysis is performed where the citation is matched with reference patterns of the given dataset. Then they find out that if the correlation factor is near 0 then the statements are not correlated and if it is nearer to 1 they are correlated.

In this paper[8 author Proposed and expanded a solitary example definite shift-or string coordinating with calculation to search out all defined events of various examples P0, P1, P2 … Pr-1, (r≥1), every one of equivalent size m, inside the content T. This set or of numerous examples is taken care of by this person grouping coordinating. The new calculation is known as a multi-design defined shift-or (MPSO) string coordinating with calculation and further expanded MPSO by utilizing the idea of super letter sets. Execution results show that by utilizing a super letter set of size s, the calculation (MPSO) is speeded up by a component of s, where s is that the size of the super letter set (for example s is that the quantity of characters handled all the while). In any case, these calculations are appropriate just example length (m) is a more modest sum than or satisfactory to word length (w) of the PC utilized (for example m ≤ w).

In [9] author proposed a viable copyright infringement location device in which they processed likeness measures by utilizing Cosine similitude and Jaccard comparability capacity to distinguish literary theft in text-based tasks. There are two sorts of approaches one is content-based and the following one is Stylometric based for understanding appropriated reports. In the substance based technique, the report is broke down utilizing semantically related words in a succession of words or vector space model, and in another strategy, they utilized linguistic style for discovering copyright infringement. Subsequently in both the techniques the primary strategy is more powerful and figures great outcomes. This paper is developed vector space model utilizing unigram, bigram, and trigram vector and afterward determined record recurrence for each term and afterward determined cosine comparability for all vectors. In the following stream, they determined the likeness score for the trigram portrayal of a succession of words utilizing the Jaccard similitude measure for each pair of an archive. Finally, they reasoned that the cosine similitude shows great outcomes than the Jaccard comparability measure and cosine closeness is more best.

In this paper author [10] proposed a method in which they have created plagiarism cases using obfuscation technique. The word "Obfuscate" is first time used in 1536 which means making things unclear, confusing, and unclarified. In obfuscating approach author tries to change the content of the document either artificially or by simulating a machine. Whenever someone tries to change the order of sentences, Part of speech, synonym substitution, inserting or deleting new terms in the given document is used to create an unintelligible case of plagiarism. It performs fragment obfuscation after this they found the degree of similarity between the sentences. After constructing the corpora the suspicious documents are compared with the source document then finally for each pair of the document, a metafile is created with some important attributes.

In this paper author [11] has proposed ML has its starting point from the creature learning hypothesis. ML doesn't need earlier information yet can independently get a discretionary strategy with the help of information got by experimentation and consistently connecting with the unique climate. Because of its qualities of self-improving and internet learning, support learning has gotten one among clever specialist's center advancements. This paper gives a presentation of support learning, talks about its fundamental model, the ideal approaches used in ML, the most support ideal arrangement that is wont to compensate the specialist including model free and model-based strategies – Temporal distinction technique, Q-learning, normal prize, conviction identical techniques, Dyna, focused on clearing, line Dyna. Finally yet not the littlest sum this paper momentarily depicts the utilizations of support learning and a couple of the more extended term research scope in ML.

In paper [12] author proposed an architecture Compute Unified Device Architecture which is introduced to handle Graphical processing units and multiple threads simultaneously. They have used this same architecture on both types of the processor – GPU and CPU. CUDA implements the algorithm in four sections to improve the performance of the system. The first two sections are responsible for the execution of the algorithm either serial or less parallel read of the instruction on the given CPU. In the later one, the parallel documents are compared with one another and compute similarity documents accordingly. After this they considered the different factors affecting the overall performance of the system such as the size of the memory block size, memory allocation type also matters to improve algorithm performance like they have used Global memory of the GPU and all the documents are stored here to accommodate the large size of documents.

In this paper [13] author proposed a disguised plagiarism detection system in Arabic text [33]. They think plagiarism detection is a challenging task for natural language and proved the same. As the Arabic language is used worldwide, there are many resources available online to plagiarist the content, therefore a difficult task. The disguised plagiarism comes under the category of rewording, paraphrasing, synonym replacement, text manipulation, etc. They propose two approaches in detecting plagiarism in Arabic text as the first approach is based on word alignment, word embedding, and word weighting to measure semantic similarity, while the other approach is based on Machine Learning (ML).

The difficulty in plagiarism detection arrives due to the high morphological nature, semantic and syntactic features of natural languages. For natural language, plagiarism can be detected either by the intrinsic approach or by the extrinsic detection approach. Here word embedding and machine learning methods are followed on extrinsic plagiarism detection on Arabic texts. The word embedding (WordEmbed) method uses word alignment, word representation, bags of meaning, part of speech tagging, frequency weighting for creating representation vector for each pair of sentences, and comparison is analyzed between suspicious and source documents. By machine learning approach, plagiarism detection accuracy is determined by including different syntactic, semantic, and lexical features in the sentences of Arabic texts.

In this paper [14] author used a Temporal difference learning method which is effective and works on the principle of supervised learning in which the prediction of training signals is predicted over the successive future time. This method is used as an external plagiarism detection system because the suspected document is compared with the overall dataset or the corpus. Plagiarism detection is very crucial to find the originality of the document. They have engaged one of the best reinforcement learning methods is Temporal Difference (TD) for quick retrieval of the information. This method used a hybrid combination of two methods- Monte Carlo and the dynamic programming method. Because of this deadly combination of these two strong methods the processing power of this technique is good and is important for improving the performance of the system. After static analysis, the total plagiarism in the whole document is 18% in the group of 10 sentences.

In this paper [15] author proposed a new similarity measure with length factor although it is simple and easy but very impactful compared with another similarity measure. Various types of similarity measures have been proposed for comparing similarity textual data or documents. This proposed algorithm is considering the length factor that means the length of the content. Then they see unique words in the given contents and similar words on both documents. In this paper they have used the given formula to calculate similarity measure:

III. Proposed Work

In the proposed methodology we will explain the Semantic Analysis with the Syntax and semantic Matching Algorithm which we have used in our project to obtain a high rate for the detection of plagiarisms up to some levels that can overcome the problem of the previous techniques. The approaches discussed in the previous solutions are limited with their text content and matching algorithm.

These pseudo code steps and flow chart presents an approach for the spam matching with its appropriate solution. This section presents an algorithm steps pseudo code and important functions, which participate to perform the algorithm process. Finding the efficient value of the parameter similarity measure and computation time is performed.

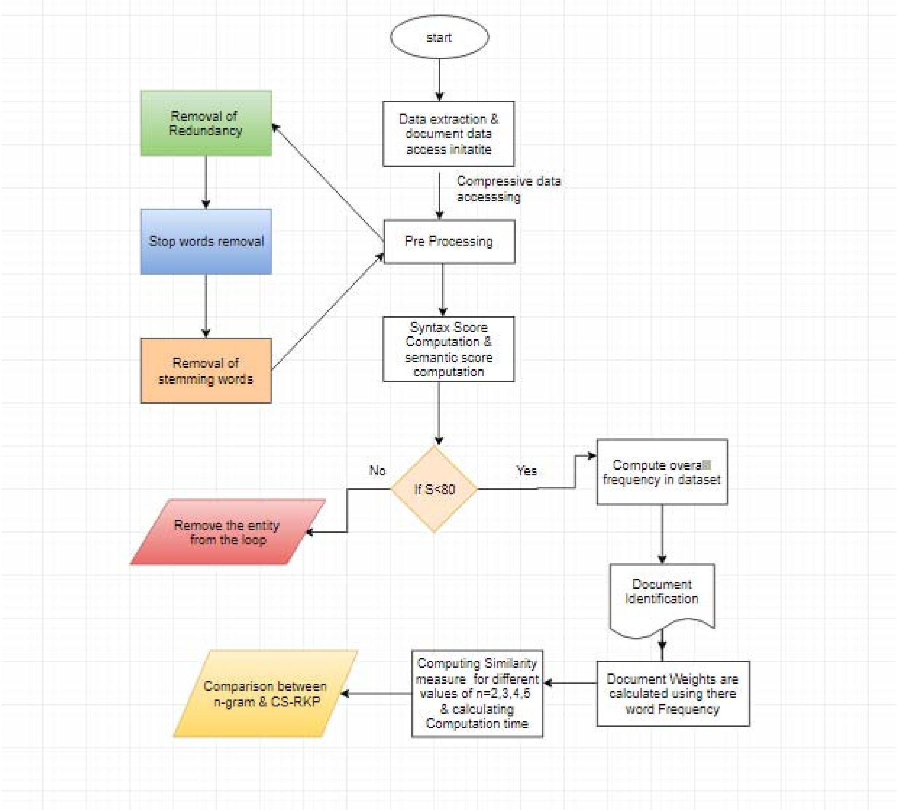

start
Data extraction & document data access initatite
Compressive data accesssing
Removal of Redundancy
Stop words removal
Removal of stemming words
Pre Processing
Syntax Score Computation & semantic score computation
No
If S<80
Yes
Remove the entity from the loop
Compute overalll frequency in dataset
Document Identification
Document Weights are calculated using there word Frequency
Computing Similarity measure for different values of n=2,3,4,5 & calculating Computation time
Comparison between n-gram & CS-RKP

Figure 2: Flow Diagram for complete execution steps.

IV. Result Analysis

The experiment is performed with simulation comparison parameters as similarity measure, computation time. Thus the comparison performance shows the efficiency of the proposed algorithm over traditional analysis.

Computation Time

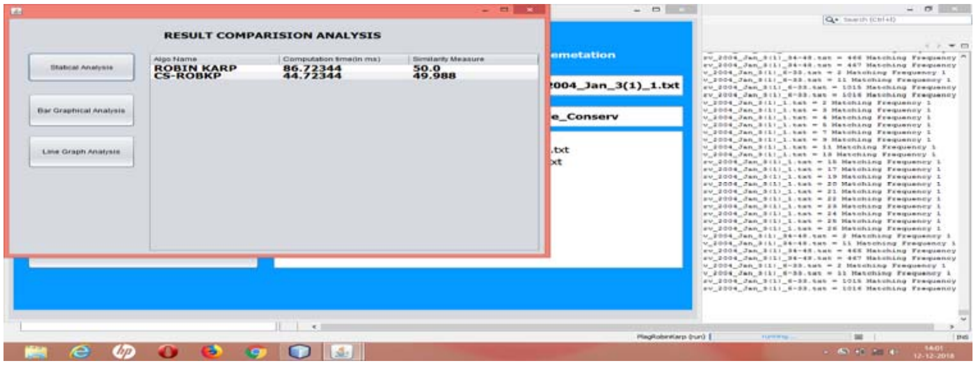
The PUBMED dataset is computed by implementing both traditional and CS-RKP algorithms defined in the tool and here as we load the dataset and verifies the eligibility and taking their features for consideration or not is the time taking process to identify and to load the images and selection of password comes under training time of a dataset, extracting the properties and making them in process format is training time.

Computation time = Final time completion – Initial time

This section deals with the execution and the framework developed are displayed over here from the initial step to the final step. The Scenario comprised of loading of the dataset, file selection, n-gram value selection, and running both the algorithms.

This section discusses provides the result for different values of N. where the value of n-gram is taken as 2, 3, 4, and 5. Thus the observation shows the efficiency of the proposed CS-RKP over the traditional N-gram approach.

Observation for N= 2:



RESULT COMPARISION ANALYSIS

Algo Name	Computation time(in ms)	Similarity Measure
ROBIN KARP	86.72344	50.0
CS-ROBKP	44.72344	49.988

Figure 3: Comparative analysis for computation time and similarity measure (n=2).

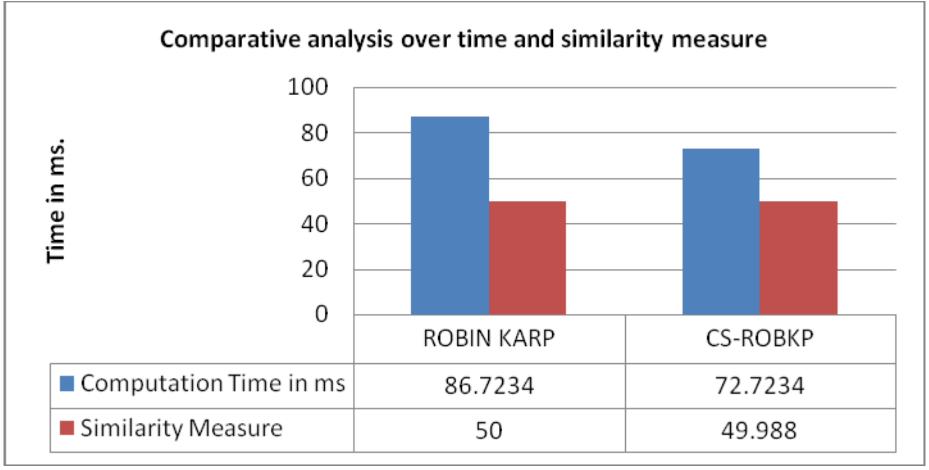
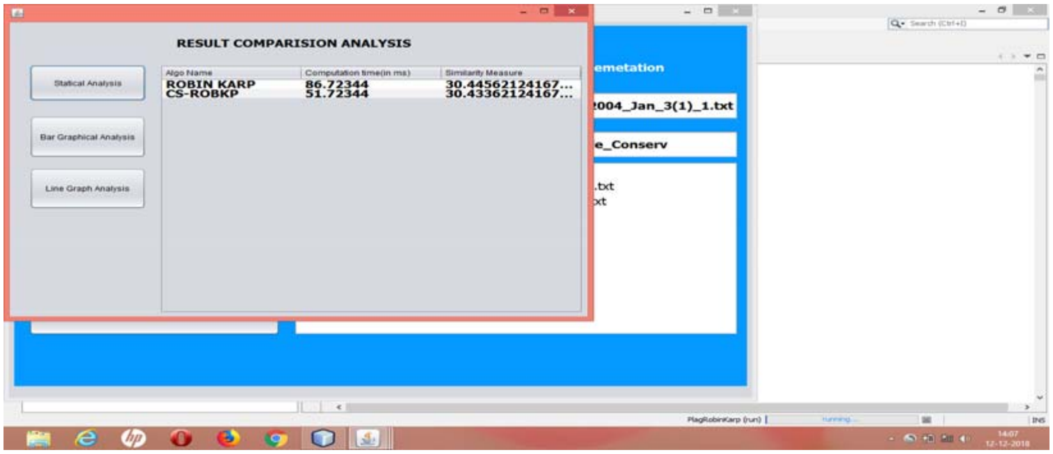


Figure 4: Bar graph representation of N-gram and CS-RKP algorithms for N=2.

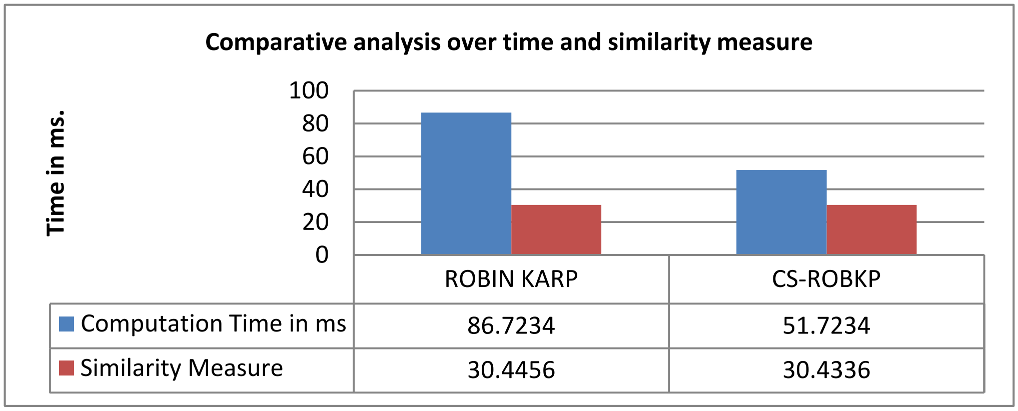
Observation N as 3:



RESULT COMPARISION ANALYSIS

Algo Name	Computation time(in ms)	Similarity Measure
ROBIN KARP	86.72344	30.44562124167...
CS-ROBKP	51.72344	30.43362124167...

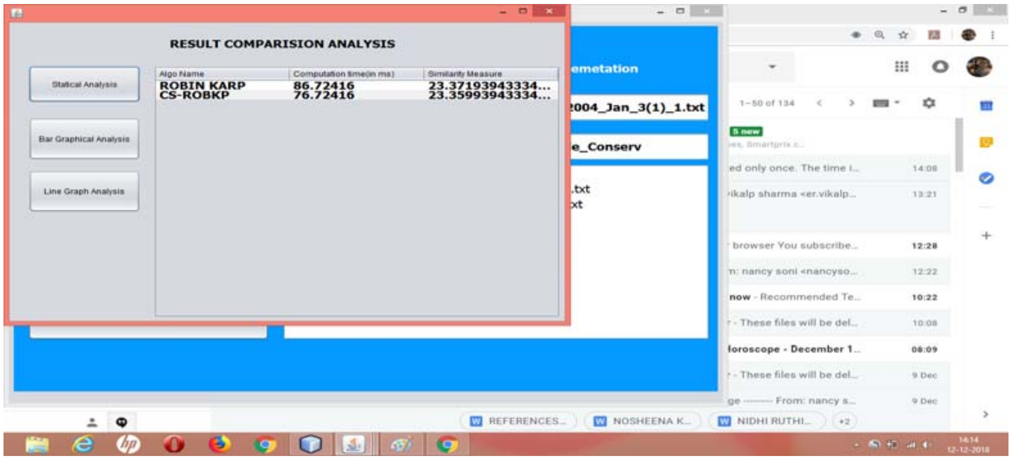
Figure 5: Comparative analysis for computation time and similarity measure (n=3)



	ROBIN KARP	CS-ROBKP
🟦 Computation Time in ms	86.7234	51.7234
🟥 Similarity Measure	30.4456	30.4336

Figure 6: Bar graph representation of N-gram and CS-RKP algorithms for N=3.

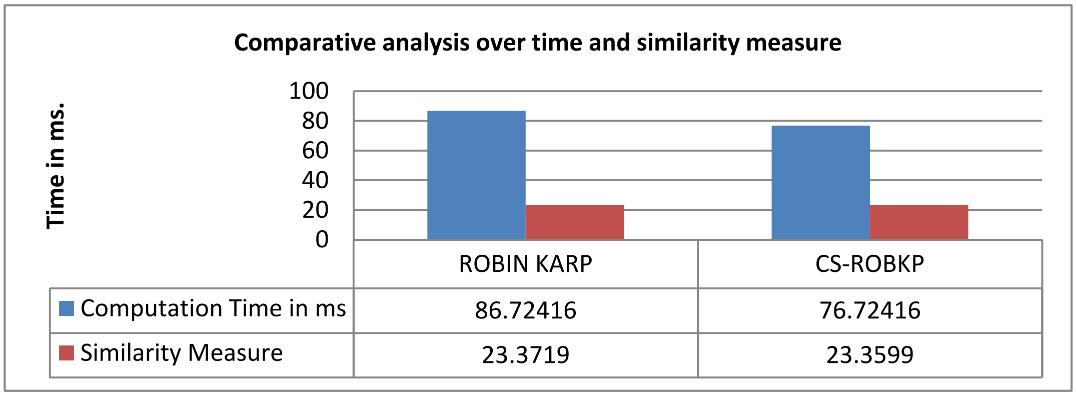
Observation N as 4:



RESULT COMPARISION ANALYSIS

Algo Name	Computation time(in ms)	Similarity Measure
ROBIN KARP	86.72416	23.37193943334...
CS-ROBKP	76.72416	23.35993943334...

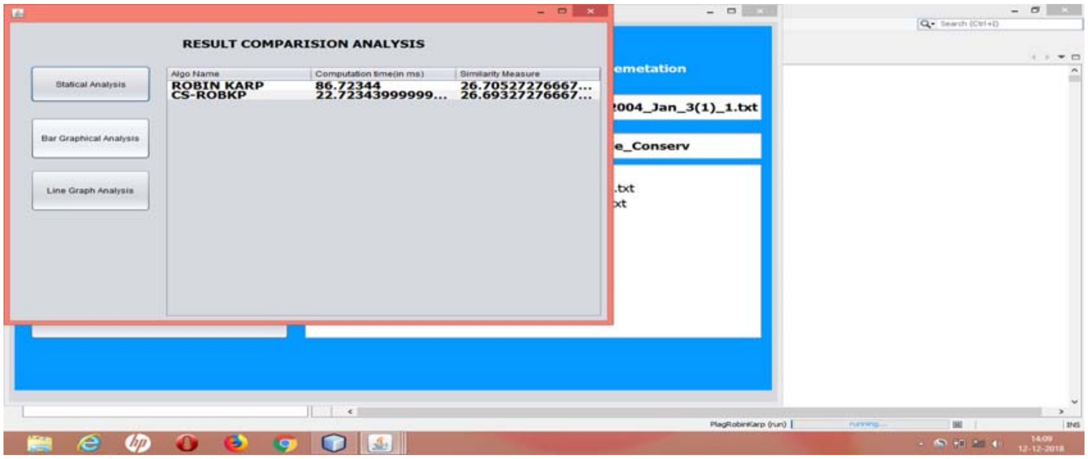
Figure 7: Comparative analysis for computation time and similarity measure (n=4)



	ROBIN KARP	CS-ROBKP
🟦 Computation Time in ms	86.72416	76.72416
🟥 Similarity Measure	23.3719	23.3599

Figure 8: Bar graph representation of N-gram and CS-RKP algorithms for N=4.

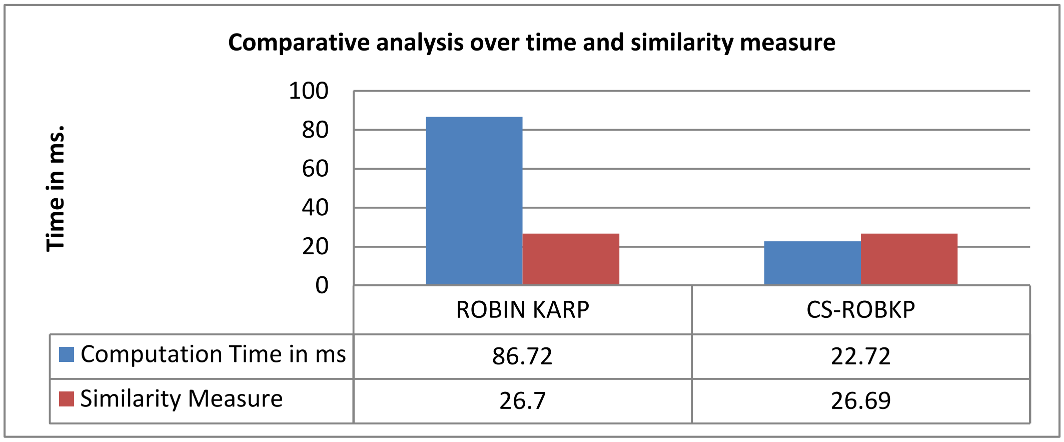
Observation N as 5:



RESULT COMPARISION ANALYSIS

Algo Name	Computation time(in ms)	Similarity Measure
ROBIN KARP	86.72344	26.70527276667...
CS-ROBKP	22.72343999999...	26.69327276667...

Figure 9: Comparative analysis for computation time and similarity measure (n=5).



	ROBIN KARP	CS-ROBKP
🟦 Computation Time in ms	86.72	22.72
🟥 Similarity Measure	26.7	26.69

Figure 10: Bar graph representation of N-gram and CS-RKP algorithms for N=5.

Table 1: Statically analysis of Similarity measure between algorithms for N=2, 3, 4, 5.

Algorithm N= Value	CS Rabin- Karp	N-GRAM
N=2	49.9	50
N=3	30.43	30.4456
N=4	23.35	23.37
N=5	26.6	26.7

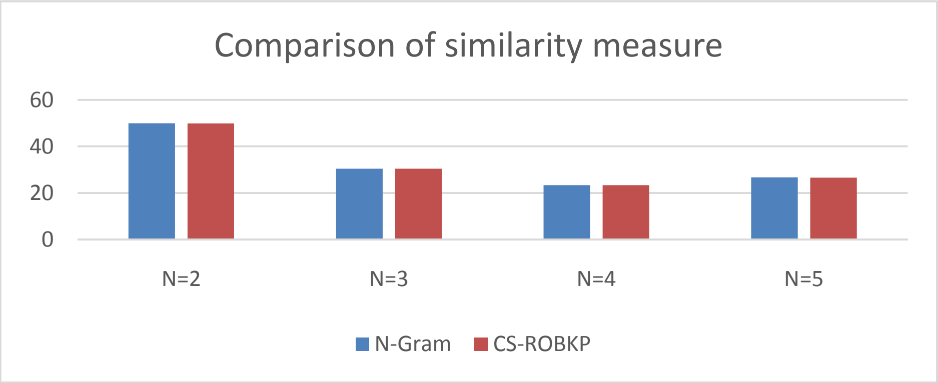


Figure 11: Bar graph representation of results at N=2,3,4,5.

The above graph shows the Bar Graph representation and efficiency of the proposed solution over similarity measure for N=2, 3, 4, 5.

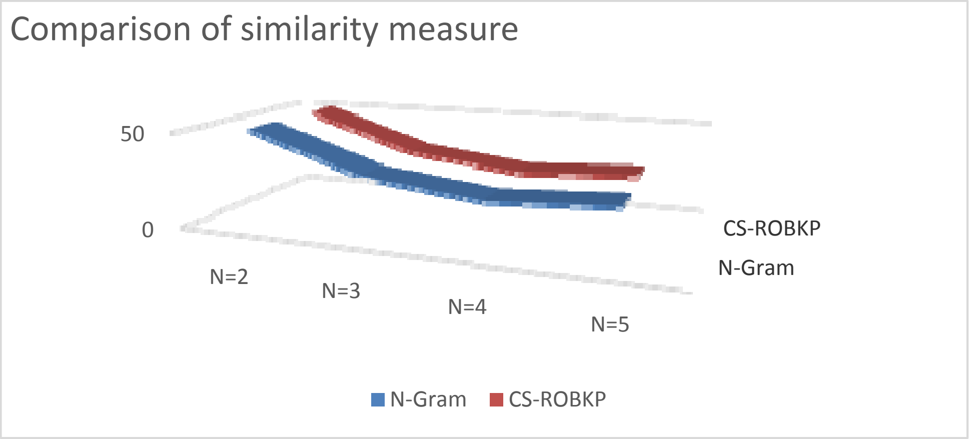


Figure 12: Line graph representation of results at N=2,3,4,5.

The above graph shows the Line Graph representation and efficiency of the proposed solution over similarity measure for N=2, 3, 4, 5.

Table 2: Comparative analysis between algorithms for N = 2, 3, 4, 5.

Algorithm N= Value	CS Rabin- Karp	N-GRAM
N=2	72.723	86
N=3	51	86.7
N=4	76	86
N=5	22.72	86

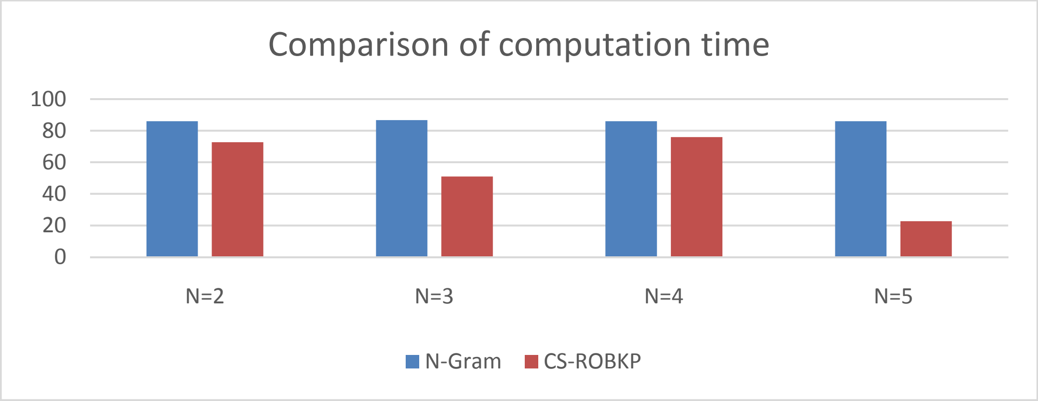


Figure 13: Bar graph representation of results at N=2,3,4,5.

The above graph shows the Bar Graph representation and efficiency of the proposed solution over Computation Time for N=2, 3, 4, 5.

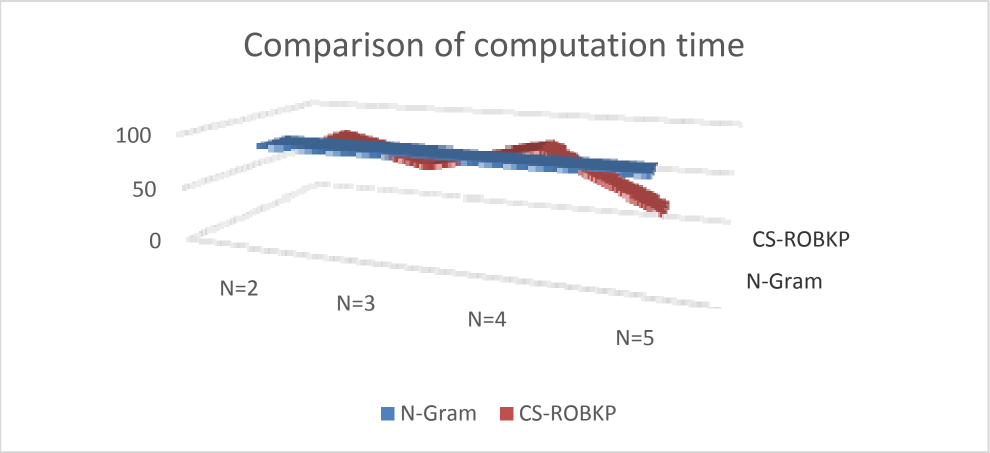


Figure 14: Line graph representation of results at N=2,3,4,5.

The above graph shows the Line Graph representation and efficiency of the proposed solution over computation time for N=2, 3, 4, 5.

V. Conclusion

Approaches which is given in past work is single approach based solution such as finding syntax-based similarity, finding the semantic-based similar word-based matching similarity, and also given hash-based similarity. The approaches are limited and good at one side of matching but lacking in producing the refined result with high throughput and similarity. Many algorithms have been proposed in the past research for plagiarism analysis over the document. To generate more efficiency for the spamming content detection, the Compressive Sensing-based RKP algorithm is defined by this research. This approach uses sampling matrix and costs function syntax and semantic computation. Finally, Rabin-Karp similarity measure evaluation is performed. The experiment is conducted using the java framework using swing API. The result computations with parameter as throughput, computation time and similarity measure shows the efficiency of the proposed algorithm over traditional analysis. The proposed solution exhibits the proven solution and is performed over the files obtained from the internet dataset stream. Thus the approach is appropriate and can be used further along.

References:

[1] Eissen S.M.., Stein B. (2006) – “Intrinsic Plagiarism Detection” in Information Retrieval. ECIR 2006. Lecture Notes in Computer Science, vol 3936. Springer, Berlin, Heidelberg.

[2] Kashkur, M. and Parshutin, S. 2010 - “Research into Plagiarism Cases and Plagiarism Detection Methods” in Scientific Journal of Riga Technical University. pp. 138-143.

[3] H. A. Maurer, F. Kappe, B. Zaka, “Plagiarism-a survey” in J. UCS 12 (8) (2006) 1050-1084.

[4] M Z Eissen, B Stein, and M Kulig – “Plagiarism detection without reference collections” In Proceeding of 30th Annual Conference of the German Classification Society pages 359–366, Berlin, 2007.

[5] AndysahPuteraUtamaSiahaan, Mesran, Robbi Rahim, Dodi Siregar- “K-Gram As A Determinant Of Plagiarism Level In Rabin-Karp Algorithm”INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 6, ISSUE 07, JULY 2017.

[6] R. Sutoyo et al., "Detecting documents plagiarism using winnowing algorithm and k-gram method," 2017 IEEE International Conference on Cybernetics and Computational Intelligence (Cybernetics.Com), Phuket, 2017, pp. 67-72. DOI: 10.1109/CYBERNETICSCOM.2017.8311686.

[7] S.N. Autade et al--- “EMAS Framework for Text Plagiarism Detection”, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 8 (2017) pp. 1584-1590.

[8] Rajesh, Prasad & S., Agarwal. (2007), “Optimal Shift-Or String Matching Algorithm for Multiple Patterns”. Lecture Notes in Engineering and Computer Science. 2167.

[9] M. A. C. Jiffriya, M. A. C. A. Jahan and R. G. Ragel, "Plagiarism detection on electronic text based assignments using vector space model," 7th International Conference on Information and Automation for Sustainability, Colombo, 2014, pp. 1-5, doi: 10.1109/ICIAFS.2014.7069593.

[10] Salar Mohtaj, Habibollah Asghari, Vahid Zarrabi –" Developing Monolingual English Corpus for Plagiarism Detection using Human Annotated Paraphrase Corpus" Notebook for PAN at CLEF 2015.

[11] Rashmi Sharma, Manish Prateek and Ashok K Sinha. Article: Use of Reinforcement Learning as a Challenge: A Review. International Journal of Computer Applications 69(22):28-34, May 2013.

[12]MAC Jiffriya, MAC Akmal Jahan Hasindu Gamaarachchi, and Roshan G. Rage- "Accelerating Text-based Plagiarism Detection Using GPUs" in December 2015. DOI: 10.1109/ICIINFS.2015.7399044

[13] El MoatezBillahNagoudi, HaddaCherroun, Ali Alshehri,-"Disguised Plagiarism Detection in Arabic Text Documents", IEEE 2018.

[14] Sudhir D. Salunkhe, S. Z. Gawali– "A Plagiarism Detection Mechanism using Reinforcement Learning" in International Journal of Advance Research in Computer Science and Management Studies Volume 1, Issue 6, November 2013.

[15] Dhruba jyoti Baruah, Anjana Kakoti Mahanta- "A New Similarity Measure with Length Factor for Plagiarism Detection" in the Proceedings of International Journal of Computer Applications (0975 – 8887) Volume 72– No.14, May 2013.