Measuring Performance of N-Gram and Jaccard-Similarity Metrics in Document Plagiarism Application

To cite this article: Nova Eka Diana and Ikrima Hanana Ulfia 2019 J. Phys.: Conf. Ser. 1196 012069

View the article online for updates and enhancements.
Measuring Performance of N-Gram and Jaccard-Similarity Metrics in Document Plagiarism Application

*Nova Eka Diana, Ikrima Hanana Ulfa
Informatics Department, Faculty of Information Technology, Universitas YARSI, Jl. Letjend Suprapto, Cempaka Putih, Jakarta Pusat, Indonesia, 10510

*nova.diana@yarsi.ac.id

Abstract. String-based similarity metrics were mainly used to lexically measure the similarity between words based on the string sequences and character compositions. This research aimed to build an application that can identify the similarity between documents. The program employed two lexical-based algorithms, N-gram and Jaccard, to check the documents similarity. The author focused on analysing the algorithms’ performance based on accuracy, sensitivity, and efficiency metric. Datasets used in this research were the final thesis documents in Indonesian and English language. Experiment results revealed that Jaccard algorithm has a better performance in term of accuracy and sensitivity compared to N-gram. Notwithstanding its superior performance, Jaccard had a longer running time than N-gram to process documents. Furthermore, the results also pointed out that the cross-language documents were indeed affecting the degree of similarity checking.

1. Introduction
Plagiarism is adopting other works as his own without providing an explicit acknowledgment. This action is considered as unscrupulous behaviour and may receive discipline if proven guilty. Nevertheless, in an academic environment, such as a university, undergraduates usually possess a limited understanding of the meaning of plagiarism. Therefore, it is impending for the students to create slips while drafting their final papers. It is also demanding and exhaustive for the supervisor to manually recognize the forgery for all students’ reports. An application that can automatically discover documents falsification of online or offline sources will benefit the supervisors and students to mitigate the plagiarism.

In computer science, text similarity measures remain the most primary and necessary task in text associated works, such as information retrieval, documents clustering, and documents summarization. Methods to discover the relationship between documents is distinctive between structured and unstructured data. In the structured documents, we can apply the features of existing DBMS to maintain and elicit information. It is likewise permissible to perform a modularity proposal to resolve the steadily updated and growing records in the database [1]. However, it is hard and challenging to attain the likeness in unstructured documents. The domain of this research was textual data; that was unorganized documents. The principal idea in textual retrieval was determining the correlation between words or sentences in reports either lexically or semantically. Lexical similarity uses a string-based metric, either with a character-based or a term-based approach, to affirming the similarity between documents. Whereas, a semantic relation between two texts apply corpus-based and knowledge-based method.
Some lexical similarity methods are LCS, N-gram, Jaccard Similarity, and Matching Coefficient. Conversely, semantic similarity techniques are Hyperspace Analogue to Language (HLA), Latent Semantic Analysis (LSA), and Explicit Semantic Analysis (ESA) [2].

Sun, Ma, and Wang (2015) in their research evaluated the performance of string-based similarity metrics based on four distinct strategies: character-based, token-based, language-based, and hybrid methods. Their study revealed that the hybrid methods were not recommended to apply if the intention was performance. Character-based and token-based methods such as N-gram, ISUB, SoftTFIDF, and Jaccard yielded the best F-measure value and the speedy running time opposed to other algorithms [3].

In this study, we developed a client-based application to classify the documents forgery. We applied String-based similarity measures, Jaccard Similarity and N-gram metrics to gauge the association degree among documents. Next, we examined the performance of both algorithms based on accuracy, sensitivity, and efficiency metrics. We arrange this paper as follows. Section 1 defines the background of the research. Section 2 describes related works important to this research. Next, Section 3 illustrates the methodologies for executing the research. Section 4 comprises the experiment outcomes and its evaluations. Finally, Section 5 exhibits the conclusion and future works of our research.

2. Related Works

According to the Code of Students Rights, Responsibilities, And Conduct at Indiana University, plagiarism was described as presenting other’s works as part of hers. Any thoughts or substances attained from other references either in written or oral should be declared with a prudish citation or reference unless such information was public knowledge [4]. Torres-Diaz, Marin-Gutierrez, and Hinojosa-Becerra investigated the advantage of technologies on the level of plagiarism in university. The result indicated that forgery tends to diminish with the better adoption of technologies [5]. Some researches had been performed to enhance the support of technologies in alleviating forgery acts [6]–[10]. Researches steadily evolve and refine the algorithms to increase the identification rate of plagiarism.

String-based similarity remains the most promising methods to detect the similarity among documents. Some of them are N-Gram, Jaccard, ISUB, and SoftTFIDF. In this research, we adopted only two approaches, N-Gram and Jaccard to build a plagiarism application. Vishwakarma et al. stated that the use of N-gram indexing in information retrieval, especially in monolingual documents, gave the increase of mean average precision value [11]. Hussein in his research processed documents into N-gram phrases and used Latent Semantic Analysis (LSA) to investigate similarities in Arabic documents. He revealed that the proposed method had robust capabilities in analyzing and visualizing the apparent and other intelligent similarities of documents [12]. Besides finding the similarities between documents, N-gram could reduce the storage space, optimize the running time, and validate sentences in documents [13], [14]. Lecluze et al. further investigated the best granularity between character N-gram and word N-gram in tackling multilingual automatic alignment. Their observation showed that the word granularity N-gram granularity was more efficient than the word granularity in solving multilingual alignment [15]. Vilares et al. [16] also confirmed that character N-gram was valid and consistent for indexing and translating purposes in Cross-Language Information Retrieval (CLIR).

Besides N-gram, many researchers had studied the applicability and effectiveness of Jaccard metric for finding documents similarities. Lu et al. [17] combined multi-dimensional similarity and Jaccard-K-means clustering to generate news recommendation. The result proved that their proposed method could enhance the scalability and the accuracy of the recommendation system despite of data sparsity. Verma, Agarwal, and Khan [18] applied the Jaccard Similarity to obtain the polarity of the item in E-commerce customers’ opinions. The research showed that the proposed method was more efficient and accurate in classifying customer opinions that the existing system. Other researchers had tried to compare the performance between MySQL pattern matching and Jaccard Similarity on query suggestion. Also, the result pointed out that Jaccard query suggestion generated more accurate result but with a longer processing time than MySQL pattern matching [19]. Ayub et al. [20] also confirmed the beneficial of Jaccard Similarity Metric. Their research suggested that the inclusion of the Jaccard measure in
recommender system had improved the collaborative filtering (CF) concerning Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).

3. Material and Methods

3.1. Application Architecture

Figure 1 describes the architecture of the proposed application. The proposed system adopted an FTP connection to communicate between server and client. The server stashed all corpus texts in pdf format and both in Indonesian and English. A user could pick either the N-Gram or Jaccard algorithm to concoct the presented document. Next, the application would estimate the similarity between the input text and the database. The application will infer plagiarism if the similarity rate is higher than 70%. Otherwise, the application will insert the document to enrich the database.

![Figure 1. Application architecture](image)

3.2. Jaccard Similarity Coefficient

Jaccard Similarity Coefficient is a String-based Similarity Measure which works on string chains and characters organization using term-based similarity procedure. Jaccard is a statistical ratio of correlations between the specimens set. Equation (1) specifies the Jaccard Similarity Coefficient for two assortments of documents as the cardinality advantage of a disjoint collection over a union set of unit documents as exhibited in equation (1). $J(A, B)$ is the Jaccard similarity coefficient, $A$ and $B$ are the tokens set for the first and second document, respectively. Figure 2 draws the organization means of the Jaccard Similarity algorithm. Given two papers, we need to evoke the contents of documents and put it to STRING1 and STRING2, respectively. Next move is splitting the document content into an assemblage of tokens and count the number of tokens in each document. The size of tokens will be utilized to compute the Jaccard Similarity Coefficient as described in equation (1).

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

3.3. N-gram Measure

N-gram is also one of String-based Similarity metrics, but it employs a character-based similarity strategy. This method converts a string or a sentence into a set of N-gram. A set of N-gram for each text will then be matched to measure the association between those data. Equation (2) illustrates the
similarity value between two documents using N-gram metric. Set 1 and Set 2 are N-grams for the first and second document, respectively. Figure 3 explains the detail process of N-gram algorithm. STRING1 and STRING2 are the input documents, while GRAM1 and GRAM2 are the grams set of each document. The method will eliminate a particular gram from the second set if the gram appears in both groups. The final set of grams is the input for computing similarity value in equation (2).

\[ sim(set1, set2) = \frac{2 \cdot |set1 \cap set2|}{|set1| + |set2|} \]  

(2)

3.4. Experiments Setup

We had a corpus of 50 pdf with the average page number of 100 pages. In this research, we compared the algorithms’ performance based on accuracy, sensitivity, and efficiency. Accuracy means the similarity value of two documents, sensitivity means the ability of the algorithms in finding the relevant documents, and efficiency measure the running time of the algorithms.

We divided the experiment into two scenarios:

a) The first outline
   This scenario aimed to evaluate the accuracy of the methods for distinguishing the same document in the database as the input. We used all texts in the database as the training data and randomly elected ten reports in the corpus as the testing data. Given an input document, we looked at the similarity state between this report and its precisely same document in the training data. If a method gave a higher similarity value; then the more accurate was an algorithm.

b) The second outline
   This experiment aimed to evaluate the efficacy of the methods in producing the raised comparison value to the relevant documents. In this experiment, we selected all 50 documents in the database as the data training and purposely selected five articles in the database as the testing data. Each document in the testing data had two relevance documents in the training data, which were its article and report document of that article.
4. Results and Discussions

The first scenario aimed to measure the accuracy of the methods in identifying the similarity between two specific documents. Figure 4 compares the performance between Jaccard and N-Gram method, with the average similarity value of 100% and 83.52%, respectively. Jaccard similarity performed better due to its consistency in iterating each word or token in both documents. Meanwhile, N-gram method did not iterate each gram in both documents and only processed some grams in the first document. If the specific gram already appeared in the first document, then the method will not the similar gram in the second documents. Additionally, the number of iterations will determine the running time of the algorithm. Jaccard needed an average of 6-18 minutes to process a document; meanwhile N-gram only took around 1-2 minutes. The higher number of iterations, then the longer time needed to check the similarity between two set of documents. N-gram metric had a lesser time because it eliminated the actions to process the same grams.

Table 1 illustrated the results of the second scenario. Jaccard algorithm had a mean average precision (MAP) value of 0.80, while N-gram had a MAP value of 0.90. N-gram wrongly identified only one relevant document; meanwhile Jaccard was incorrectly identifying the relevant documents in two out of five experiments. This result was affected by the mixed language (Indonesian and English) in the document contents. Both Jaccard and N-Gram were string-based algorithms. Hence, the contents and languages of a document will affect the similarity value.

![Jaccard Similarity Value vs N-gram Similarity Value](image1)

![Jaccard Similarity - Running Time vs N-gram Running Time](image2)

**Figure 4.** Jaccard vs N-Gram (Accuracy)
Table 1. Sensitivity Results

| Doc No | Jaccard Similarity | N-gram |
|--------|---------------------|--------|
|        | 1st Doc (%) | 2nd Doc (%) | T (m) | 1st Doc (%) | 2nd Doc (%) | T (minute) |
| DOC-1  | AR-1 (100) | REP-1 (43,5) | 6 | AR-1 (83,89) | REP-1 (45,57) | 1 |
| DOC-2  | AR-2 (100) | AR-3 (9,84) | 5 | AR-2 (81,76) | REP-2 (25,18) | 2 |
| DOC-3  | AR-3 (100) | AR-2 (9,84) | 5 | AR-3 (85,27) | AR-2 (22,80) | 1 |
| DOC-4  | AR-4 (100) | REP-4 (27,5) | 5 | AR-4 (84,72) | REP-4 (32,25) | 1 |
| DOC-5  | AR-5 (100) | REP-5 (30,3) | 5 | AR-5 (80,73) | REP-5 (35,53) | 1 |

5. Conclusion and Future Works
We implemented Jaccard Similarity and N-gram to develop an application for detecting plagiarism in documents. We compared the algorithms’ performance based on three criteria: accuracy, relevance, and efficiency. Experiment results showed that Jaccard Similarity has a better accuracy performance in finding the relevant documents. Contrariwise, Jaccard needed a longer time to calculate the similarity value compared to the N-gram algorithm. In conclusion, the number of iteration in similarity checking algorithm determined the running time and also the accuracy of the method. The cross languages in documents set were also affecting the result of similarity checking. Parallel or multi-threading process may be applied to improve the running time of the Jaccard Similarity. Translation-approach may be considered to target corpus documents in the different language with the input.

6. References
[1] Yuliwulandari R, Diana N E and Prayuni K 2017 The First Modular Database of Indonesian Genes-Associated Diseases Information J. Theor. Appl. Inf. Technol. 95 pp. 1499-1508
[2] Gomaa W H and Fahmy A A 2013 A Survey of Text Similarity Approaches Int. J. Comput. Appl. 68 13-18
[3] Sun Y, Ma L and Wang S 2015 A Comparative Evaluation of String Similarity Metrics for Ontology Alignment J. Inf. Comput. Sci. 12 957-64
[4] Bloomington I U 2014 Definition of Plagiarism: How to Recognize Plagiarism, School of Education, Indiana University at Bloomington https://www.indiana.edu/~plag/definition.html (2018)
[5] Torres-Diaz J C, Marin-Gutierrez I and Honojosa-Becerra M 2018 Uses of internet and levels of plagiarism 13th Iberian Conference on Information Systems and Technologies (CISTI) pp 1-6
[6] Wang S, Qi H, Kong L and Nu C 2013 Combination of VSM and Jaccard coefficient for external plagiarism detection International Conference on Machine Learning and Cybernetics pp 1880-85
[7] Baba K, Nakatoh T and Minami T 2017 Plagiarism detection using document similarity based on distributed representation Procedia Comput. Sci. 111 382-7
[8] Sutoyo R et al 2017 Detecting documents plagiarism using winnowing algorithm and k-gram method IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom) pp 67-72
[9] Margarov G, Tomeyan G and Pereira M J V 2017 Plagiarism detection system for Armenian language Computer Science and Information Technologies (CSIT) pp 185-9
[10] Nagoudi E M B, Cherroun H and Alshehr A 2018 Disguised plagiarism detection in Arabic text documents 2nd International Conference on Natural Language and Speech Processing (ICNLS) pp 1-6
[11] Vishwakarma S K, Lakhtaria K I, Bhatnagar D and Sharma A K 2015 Monolingual Information Retrieval using Terrier: FIRE 2010 Experiments Based on N-gram Indexing Procedia Comput. Sci. 57 815–820
[12] Hussein A S 2016 Visualizing document similarity using n-grams and latent semantic analysis SAI Computing Conference (SAI) pp 269–279
[13] Kadwe S S and Ardhapurkar S 2017 Implementation of PDF crawler using boolean inverted index and n-gram model International Conference on Wireless Communications Signal Processing and Networking (WiSPNET) pp 2680–2683
[14] Jain K, Khatri P and Indolia G 2015 Chunked N-Grams for Sentence Validation Procedia Comput. Sci. 57 209–213
[15] Lecluze C, Rigouste L, Giguet E and Lucas N 2013 Which Granularity to Bootstrap a Multilingual Method of Document Alignment: Character N-grams or Word N-grams? Procedia - Soc. Behav. Sci. 95 473–481
[16] Vilares J, Vilares M, Alonso M A and Oakes M P 2016 On the feasibility of character n-grams pseudo-translation for Cross-Language Information Retrieval tasks Comput. Speech Lang. 36 136–164
[17] Lu M, Qin Z, Cao Y, Liu Z and Wang M 2014 Scalable news recommendation using multidimensional similarity and Jaccard–Kmeans clustering J. Syst. Softw. 95 242–251
[18] Verma P K, Agarwal S and Khan M A 2017 Opinion mining considering roman words using Jaccard similarity algorithm based on clustering 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT) pp 1–4
[19] Rinarterha K and Suryasa W 2017 Comparative study for better result on query suggestion of article searching with MySQL pattern matching and Jaccard similarity 5th International Conference on Cyber and IT Service Management (CITSM) pp 1–4
[20] Ayub M, Ghazanfar M A, Maqsood M and Saleem A 2018 A Jaccard base similarity measure to improve performance of CF based recommender systems International Conference on Information Networking (ICOIN) pp 1–6

Acknowledgments
Authors wishing to acknowledge YARSI Foundation for their financial support through the YARSI Research Grant Program and YARSI Library’s staff for providing the dataset.