Volume 9, Issue 05, May 2024 ISSN (online): 2456-0006

International Journal of Science Technology Management and Research

Available online at: www.ijstmr.com

A Review on Duplication Detection tool for Articles

Janhavi P. Jadhav
MSC (Computer Science)
Indira College, Malegaon

Prof. T. P. Sharma
MSC (Computer Science)
Indira College, Malegaon

Abstract-: ***Electronic media has been developing rapidly nowadays, resulting in a large number of news articles produced online, and thus duplication detection is needed. Besides, articles duplication is directly related to articles plagiarism. We explore the duplication detection in news articles from the newest online We Media data. Nowadays, due to the rapid development of electronic media and the large number of articles created online, duplication detection is required. Additionally, there is a clear connection between plagiarism in article content and the duplication of articles. Newspaper articles have been the subject of most of the previous research on duplicate detection. To validate our proposed approach, tool is meant to crawl huge number of articles data for detection. Moreover, it will apply our approach to detect plagiarism articles based on our duplication results in future. Further with the help of tool, it will try to conduct an empirical study and summarize a few of most used plagiarism patterns in plagiarism articles.***

Keywords-: ***Duplicate detection, articles, plagiarism detection***

I. INTRODUCTION

Duplication detection is very useful for a variety of tasks (e.g., file management, copyright protection and plagiarism prevention). However, existing works about duplication detection mainly focus on documents or code duplication detection, and only a few works aim at duplication detection in news articles. Moreover, existing works just detect and analyze duplication in newspaper articles [1].

Previously, only press agencies could publish news articles, but now the advent of we media makes everyone can publish and share news online. There lacks duplication detection and analysis based on the newest We Media data. Hence, we focus on the study of duplication detection in news articles from We Media which contain larger number of data. With the rapid development of electronic media, especially the We Media [2], such as Weibo, WeChat and Toutiao, it enables rapid publishing and instant access to news articles. Meanwhile, a large number of news articles are produced online, and articles duplication can no longer be ignored, since duplicate articles will increase the redundancy and management costs. Moreover, articles duplication is directly related to articles plagiarism. In the field of journalism, plagiarism is considered a breach of journalistic ethics, so it is very important to detect duplication in news articles.

II. LITERATURE SURVEY

The previous work on near-duplicate detection has focused on judging the similarity of two documents without performing a computationally-expensive bit-wise comparison of entire documents. Detection of identical documents can be done by hashing the documents and comparing the hash values: any documents with matching hashes are then considered to be duplicates. This method is not applicable to near-duplicate detection however, as it provides no information about how similar any two candidates are to one another. For near-duplicate detection, most methods create a compact representation of a document and use these for comparison, though they vary in how representations are formed and compared.

The shingling method (Broder, 2019) views a document as a set of overlapping n-grams (or shingles, short sequences of words in the text). The similarity of two documents can then be measured by calculating their set similarity. When this value is above a given threshold, documents can be considered near-duplicates of one another.

This method is costly however, as the number of shingles generated can be quite large. To deal with this, Broder proposes generating a document sketch either by finding the set of minimum hash values of a random sample of shingles within a document with a set of permutations, or eliminating all of the shingles where, for a given shingle S and number m, S mod m 6= 0.

Another method of determining similarity between documents is cosine similarity (Salton et al., 2021). This method translates documents into vectors, where each element of the vector relates a term's importance in the corpus (generally, the elements are calculated with term frequency – inverse document frequency (TF-IDF) scores). Similarity between two documents is then expressed as the cosine distance between their vectors.

Charikar (2022) introduced locality-sensitive hashing to estimate similarity between documents without the memory overhead needed to keep full vector representations of each document in memory. To do this, Charikar proposes the use of hashing functions where the similarity between two documents is the probability that their hash values are equal. Each document is then hashed with some number t of these functions, and the resulting hash values are placed in a vector which represents the document.

If two documents share a number of matching vector elements that surpasses a given threshold, they are considered near-duplicates.

Chowdhury et al. (2022) introduced I-Match, an algorithm that generates lightweight document fingerprints by hashing all of the significant tokens present in a document.
Then, whenever two documents share a fingerprint, they can be classified as near-duplicates. This method is sensitive to very slight changes in document content. Kołcz et al. (2004) presented a variation where n variants of the lexicon are used. In each variant, some portion of the lexicon is dropped, and a fingerprint is generated by hashing the document once for each variant of the lexicon. This results in each document being represented by a vector of hash values. When testing two documents for near-duplicate status, if any of the matching pairs of elements of their vectors collide, the documents can be considered near-duplicates.
Henzinger (2021) proposed detecting near-duplicates by combining shingling and locality sensitive hashing. The method begins by using shingling to identify near-duplicate candidates, and then filters these results using locality sensitive hashing to identify near-duplicate documents. The author obtained promising results on a very large dataset, both for near-duplicate documents on the same website and near-duplicate documents on different websites.

III. SYSTEM ARCHITECTURE

The entire process of our proposed approach is summarized in Fig. 1. It can be considered in two phases: (1) Normalization; and (2) Detection. The following subsections describe the detailed design of each phase

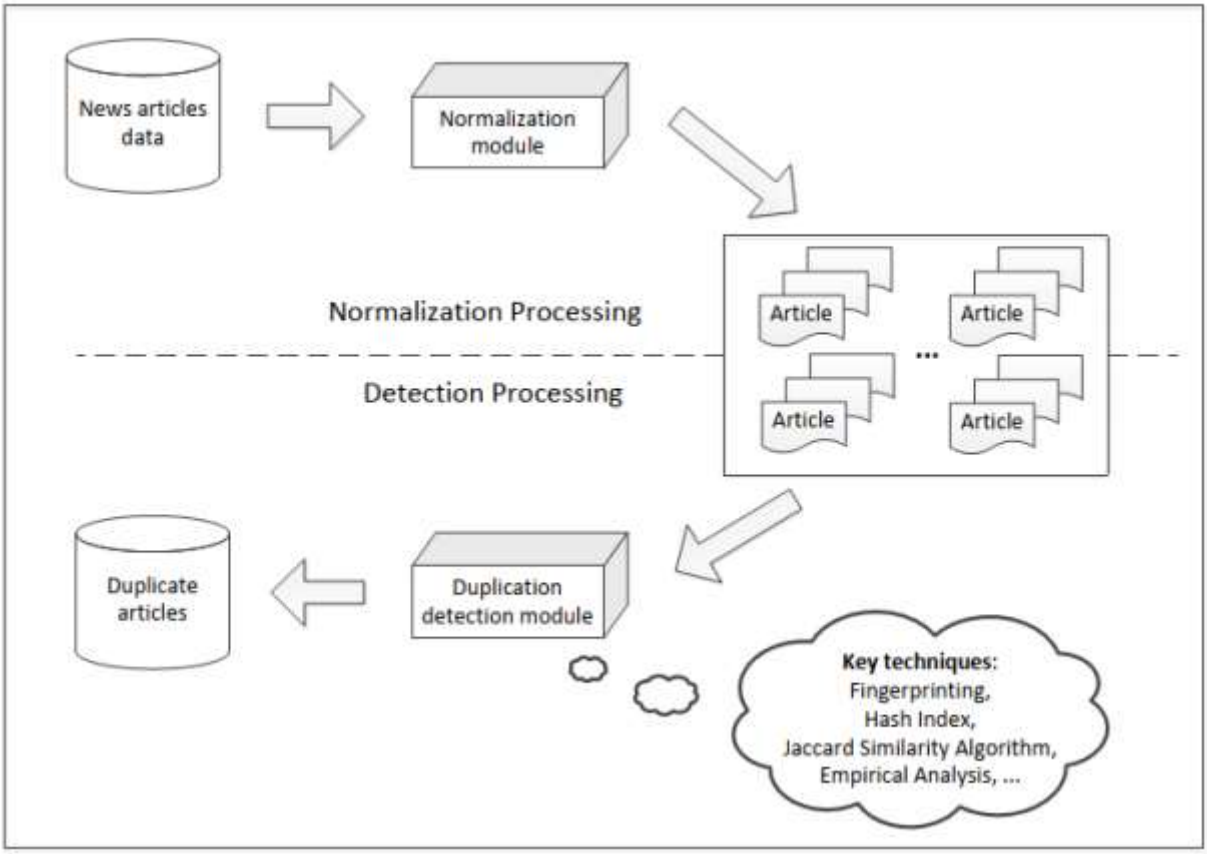

Figure 1. Detail System architecture

A. Normalization

In the normalization processing phase, key components of each article are first extracted from the news articles data, which are organized as the characteristic matrix: <Title, Abstract, Content, AuthorId, AuthorName, PublicTime, CommentNum, Category, Link>, where each element is a characteristic vector containing corresponding information of all articles. Although we aim at the duplication detection of Content in news articles, other components of the articles are also important, such as Title, AuthorName and PublicTime, since such information can further guide the empirical analysis on news articles plagiarism. We focus on the duplication detection of articles content, and thus we further process and normalize the content of each article. Based on the observation, we find that plagiarists often changed the position of images in the article to avoid plagiarism detection. Therefore, we remove the images in articles' content and

retain the text data for duplication detection. Furthermore, in order to reduce the redundancy of the text, we remove spaces in the text. Next, pretty-printing is used to layout the text for one sentence per line. We use single sentence as our basic unit for duplication detection. After the normalization processing phase, the original news articles are transformed into normalized text data sentence by sentence, as the input of following duplication detection phase.

B. Detection

Given a set of normalized text data, we use fingerprinting algorithm combining with hash technique to process each article's data. We evaluate pairs of articles' text are similar by measuring their ratio of matched hash values (i.e., matched sentences), and report those satisfying the similarity threshold.

CONCLUSION

We have presented an article duplication detecting technique, and implemented proposed system. We find that the top three topics with the highest proportion of duplication are Sports news, Military news, and Technology news.

REFERENCE

1. S. Bowman and C. Willis, “We media,” How audiences are shaping the future of news and information, 2022.
2. W. Kienreich, M. Granitzer, V. Sabol, and W. Klieber, “Plagiarism detection in large sets of press agency news articles,” 17th IEEE International Workshop on Database and Expert Systems Applications, 2022, pp. 181–188.
3. T. A. Van Dijk, “News analysis: Case studies of international and national news in the press,” Routledge, 2021.
4. L. Lloyd, D. Kechagias, and S. Skiena, “Lydia: A system for largescale news analysis,” In International Symposium on String Processing and Information Retrieval, 2005, pp. 161–166.
5. I. Flaounas, O. Ali, M. Turchi, T. Snowsill, F. Nicart, T. De Bie, and N. Cristianini, “NOAM: news outlets analysis and monitoring system,” Proc. SIGMOD, 2011, pp. 1275–1278.
6. M. Bautin, C. B. Ward, A. Patil, and S. S. Skiena, “Access: news and blog analysis for the social sciences,” Proc. WWW, 2010, pp. 1229– 1232. A. Austin. 2016. Murmurhash hash functions. [Online]. Available: https://github.com/aappleby/smhasher/
7. S. Niwattanakul, J. Singthongchai, E. Naenudorn, and S. Wanapu, “Using of Jaccard coefficient for keywords similarity,” Proc. IMECS, 2013, pp. 380–384.
8. A. S. Bin-Habtoor and M. A. Zaher, “A survey on plagiarism detection systems,” International Journal of Computer Theory and Engineering, 2012, 4(2): pp. 185–188.
9. S. M. Alzahrani and N. Salim, “Plagiarism detection in Arabic scripts using fuzzy information retrieval,” In Student Conf. Res. Develop., Johor Bahru, Malaysia, 2008, pp.281–285.
10. C. Liu, C. Chen, J. Han, and P. S. Yu, “GPLAG: detection of software plagiarism by program dependence graph analysis,” Proc. SIGKDD, 2006, pp. 872–881.
11. P. Wang, J. Svajlenko, Y. Wu, Y. Xu, and C. K. Roy, “CCAligner: a token based large-gap clone detector,” Proc. ICSE, 2018, pp. 1066– 1077
12. L. Lu and P. Wang. Duplication and plagiarism detection results. [Online]. Available:http://home.ustc.edu.cn/%7Ewpc520/data.rar