JUSTIN
Jurnal Sistem dan Teknologi Informasi
p-ISSN : 2460-3562 / e-ISSN : 2620-8989
DOI: 10.26418/justin.v12i1.71224
Vol. 12, No. 1, Januari 2024

Optimasi Rabin Karp dengan Rolling Hash dan k-Gram pada Similarity Check Dokumen Abstrak Jurnal

Siti Yuliyanti[a1], Euis Nur Fitriani Dewi[a2], Andi Nur Rachman[a3], Rizky[b4]
[1,2,3] *Informatika, Fakultas Teknik Universitas Siliwangi*
Jalan Kahuripan No. 24 Siliwangi Tasikmalaya, Jawa Barat
[4]*Teknik Informatika, STMIK Bandung*
Jalan Cikutra No. 113 Bandung, Jawa Barat
[1]sitiyuliyanti@unsil.ac.id
[3]andi@unsil.ac.id
[2]euis.nurfitriani@unsil.ac.id
[4]rizky@stmik-bdg.ac.id

Abstrak

Rolling hash digunakan untuk mengatasi masalah hash pada Rabin Karp dengan memperbaharui kemunculan *string* berulang dengan menghitung nilai hash dari substring, dimana nilai hash dihitung lebih cepat dengan nilai *hash* lama sehingga dapat dibandingkan secara konstan. Kemudian penelitian ini menambahkan k-gram untuk melakukan pergerakan dari kata satu ke kata didepannya, dengan tujuan mampu meningkatkan akurasi dengan pengechekan teks pada dokumen yang lebih spesifik. Tahapan penelitian meliputi pengumpulan dataset yang berasal dari dokumen abstrak jurnal yang kemudian dilakukan praproses mulai dari *cleansing, case folding, filtering, stemming* dengan *stopword* selanjutnya implementasi *rolling hash* dan *k-gram* pada Rabin Karp untuk meningkatkan sensifitas pada similarity check serta mengetahui peningkatan presentase ketepatan dalam mendeteksi kemiripan dokumen. Hasil penelitian Hasil pengujian menunjukan k=7 menunjukan kemiripan lebih tinggi dibanding k=5 karena penelitian ini menggunakan panjang karakter jurnal > n dengan nilai kemiripan tertinggi yaitu k=7 pada dokumen jurnal 4 yaitu kemiripan abstrak 49,93% dan kemiripan judul 14,00% sedangkan untuk k=5 yaitu 12,01% kemiripan abstrak dan 4,17% kemiripan judul sehingga k-gram, basis, dan modulo berpengaruh terhadap perhitungan *similarity* dokumen.

Kata kunci: *k-gram, rolling hash, rabin karp, similarity check, substring*

Optimization of Rabin Karp with Rolling Hash and k-Gram on Similarity Check of Journal Abstract Document

Abstract

Rolling hash is used to overcome the hash problem in Rabin Karp by updating repeated string occurrences by calculating the hash value of the substring, where the hash value is calculated faster with the old hash value so that it can be compared constantly. Then this study adds k-grams to move from word to word in front of it, with the aim of being able to improve accuracy by checking text on more specific documents. The stages of the research included collecting datasets from journal abstract documents which were then pre-processed starting from cleansing, case folding, filtering, stemming with stopwords then implementing rolling hash and k-gram on Rabin Karp to check journal abstract similarity. The results of the test show that k=7 shows a higher similarity than k=5 because this study uses journal character length > n with the highest similarity value, namely k=7 in journal documents 4, namely abstract similarity of 49.93% and title similarity of 14.00 % while for k = 5, namely 12.01% abstract similarity and 4.17% title similarity so that k-grams, basis, and modulo affect the calculation of document similarity.

Keywords: *k-gram, rolling hash, rabin karp, similarity check, substring*

I. PENDAHULUAN

Kemiripan dokumen sering terjadi, manakala sumber referensi yang digunakan sama, sehingga tingkat plagiarisme menjadi tinggi, hal tersebut mentrigger penggunaan rabin karp dalam *similarity check* dokumen. Namun terdapat kekurangan pada implementasinya sehingga diperlukan tambahan metode yaitu rolling hash dan k-gram pada penggunaan rabin karp sehingga pendeteksian untuk similarity chek dapat lebih akurat dalam menghadapi kombinasi kata. Langkah alternatif untuk mengetahui kemiripan pada dokumen, tentu identik

Submitted 26-09-2023; Revised 06-01-2024; Accepted 31-01-2024

dengan penjiplakan, yaitu mengambil hasil karya orang lain dengan tidak menngunakan referensi ke sumber asli [1]. Beragam software untuk mendeteksi kemiripan dokumen yang sudah ada seperti Turnitine, Eve2, WordCheck, Moss dan masih banyak lagi namun berbayar [2]. Algoritma Rabin Karp mengadopsi metode hash dalam pencarian kata, namun sering terjadi permaslahan serta sangat efektif bila digunakan untuk pencarian jamak [1]. Konsep dasar Rabin-Karp membandingkan nilai hash inputan dan substring pada teks, jika mirip dibandingkan perbandingan kemudian dengan setiap karakternya. Beberapa penelitian lebih detail membahas modulus atau modulo, k-gram dan basis untuk mrngutraikan lebih jelas tentang pengaruh terhadap implementasi pada rabin karp seperti modulo berpengaruh pada waktu proses dimana semakin kecil k-gram maka akuirasi pengechek kemiripan dokumen menjadi lebih naik [3]. Penyelesaian dalam menemukan kemiripan dokumen tentu memiliki banyak macamnya, yaitu *Fingerprinting* memiliki prinsip kerja dengan menggunakan *hashing* untuk menangani kelemahan Rabin Karp [4].

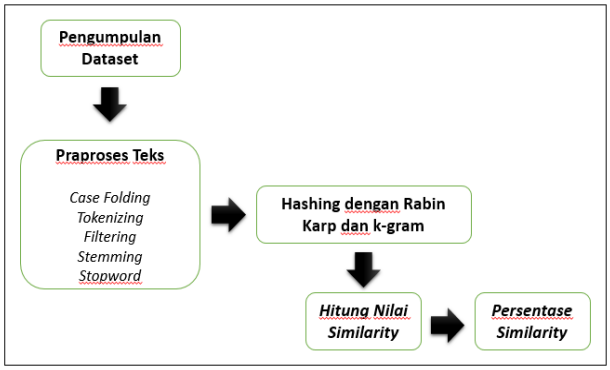
Praproses dalam sebuah penelitian tentu menjadi dasar peningkatan akurasi, semakin bersih atau baik dataset yang digunakan maka semakin mudah saat implemntasi algoritma penyelesain dalam sebuah kasus [1].

Hasil kemiripan tertinggi diperoleh dari penelitian dengan objek 10 dokumen, k=1, terbesar yaitu 57.14% dan yang terkecil sebesar 28.57% [5]. Maka penelitain ini mngadopsi hasil tersebut sebagai rujukan penelitian [6].

Perhitungan efisien nilai *hash substring* ketika melakukan geser string merupkan kunci inti, sehingga implementasi penelitian ini mampu mengoptimasi Rabin Karp serta menaikan nilai ketelitian pada saat *similarity check* dokumen [1]. Tujuan penelitian ini mampu mengimplementasikan pemodelan Rabin Karp yang dioptimasi dengan k-*gram* dan *rolling hash* pada *similarity check* dokumen abstrak jurnal agar dapat menaikan nilai akurasi dan mengetahui hasil penggunakan modulo, basis dan k-gram pada pengechekan kemiripan dokumen yang lebih spesifik.

II. METODOLOGI PENELITIAN

Tahapan penelitian diawali dengan pengumpulan dataset, bebrapa langkah dalam praproses, kemudian implementasi *Rabin Karp* dengan *rolling hash* dan *k-gram* dalam menghitung kesamaan antar dokumen dnegan melibatkan nilai modulo, basis serta k-gram [2]. Selanjutnya analisis evaluasi dari hasil presentase similarity yang diperoleh sebagaimana diilustrasikan pada Gambar 1.



Gambar 1. Kerangka Penelitian

A. *Pengumpulan Dataset*

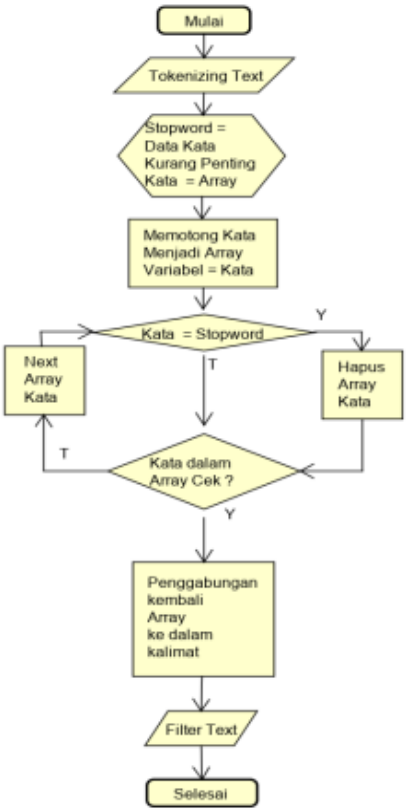
Penelitian ini menggunakan data abstrak jurnal sebagai objek penelitian untuk mengchek kemiripan dokumen. Sebanyak 30 abstrak data jurnal yang digunakan, dimana kata dalam abstrak tidak lebih dari 150 kata.

B. *Praproses Data*

Tahapan praproses data dilakukan, berdasarkan penelitian sebelumnya dengan praproses dapat meningktkan nilai akurasi dan mengurangi waktu proses terhadap data [2]. Maka tahapan pertama adalah *case folding* yaitu mengubah semua kata yang berada pada abstrak menjadi huruf kecil.

Tokenizing merupakan langkah selanjutnya, dimana setiap kata dalam abstrak dipecah menjadi perkarakter dengan menghapus *delimiter* seperti koma atau titik atau titik dua[7], kemudian melakukan *filtering* sebagaimana diilustrasikan pada Gambar 2. Tahapan *filtering* dimulai dari dataset hasil *tokenizing* kemudian dicek dengan *stopwords* dimana kata yang tidak penting di *delete* kemudian memotong kata menjadi array [6], dimana jika merupakan stopword maka dihapus jika tidak maka masuk kedalam array kata setelah selesai melalui tahapan integrasi kembali dalam kalimat menjadi dataset filtering untuk dilanjutkan ketahapan *stemming*[8].

Selanjutnya *stemming* yaitu mencari kata dasar dari setiap kata hasil dari *filtering* kemudian disimpan dalam file data praproses untuk kemudian dilanjut ke tahapan *hashing* dengan *Rolling Hash* dengan tujuan memperbaiki kekurangan yang ada pada Rabin Krap[9].



Gambar 2. Flowchart *filtering*

C. *Rabin Karp, Rolling Hash dan k-Gram*

Rabin karp adalah algoritma *string matching* dengan konsep penelusuran kesamaan pattern dalam sebuah text dengan text pembanding memalui penggunaan *hashing*[10]. Adapun tahapan Rabin karp sebagaimana diilustrasikan pada Gambar 3. *K -Gram* adalah pengolahan bahasa pada *text mining* dengan menggerakkan setiap string kedepan, dimana karakter akan dikelompokan sejumlah k. Misal kata "JURNAL", diperoleh k-gram karakter yaitu:

Unigram: J, U, R, N, A, L
Bigram: _J, JU, UR, RN, NA, AL
Trigram: _JU, JUR, URN, RNA, NAL, AL_, L_

Sedangkan jika dalam sebuah kalimat, contoh "Saya selalu berangkat kekampus pagi hari"
Unigram: saya,selalu, berangkat, kekampus, pagi, hari
Bigram: saya selalu, selalu berangkat, berangkat kekampus, kekampus pagi, pagi hari
Trigram: saya selalu berangkat, kekampus pagi hari

Gambar 3. Tahapan Rabin Karp

Hashing digunakan untuk tranformasi *string* menjadi nilai unik dengan fixed length sebagai penanda. Pendeteksian kemiripan dokumen menggunakan teknik *hash* [11], [12] digunakan untuk menemukan nilai *hash* dari rangkaian grams, pada penelitian ini *menggunakan rolling hash* karena metode ini fitur yang menghitung nilai *hash* tanpa mengulangi seluruh *string* dan nilainya numerik dibentuk dari kode *ASCII* [1] sebagai mana Persamaan 1.

$$H_{(c2\ldots cn)} = c_1 * b^{(n-1)} + c_2 * b^{(n-2)} + \ldots + c_{(n-1)} * b^{(n)} + c_{1n} \quad (1)$$

c = nilai karakter yang berasal dari kode ASCII
b = basis bilangan prima (tidak ditentukan)
n = jumlah atau panjang karakter *n-gram*

Selanjutnya menghitung kemiripan pada Rabin Karp dengan Dice's Similarity Coeficients menggunakan Persamaan 2 [13].

$$S = \frac{2C}{A+B} \quad (2)$$

S : Kemiripan
A: Total *k-Gram* dokumen 1
B: Total *k-Gram* dokumen 2
C: Total k-Gram yang sama pada A dan B

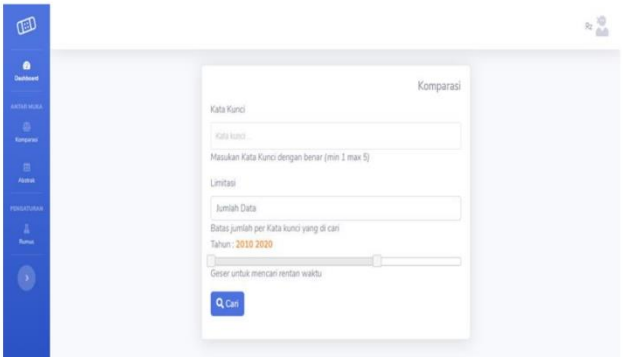
Pengujian perangkat lunak ini menggunakan metode pengujian *extreme programming* (XP) [14]. Pengujian ini berfokus pada fungsional perangkat lunak yang dibangun agar implementasi algoritma yang digunakan dapat terukur dari mulai pencarian multi dokumen abstrak jurnal dengan hasil rumus yang bervariasi, perhitungan jumlah karakter setiap dokumen abstrak jurnal, dan mengihtung persentase kemiripan dokumen yang telah dibandingkan.

III. HASIL DAN PEMBAHASAN

Tahapan penelitian setelah diperoleh hasil pengujian kemiripan judul dan abstrak jurnal dari 10 jurnal yang memiliki nilai kemiripan hanya 7 jurnal, berdasarkan Persamaan 1 dan 2, dimana semua pengujian menampilkan hasil dari pengaruh basis, k-gram dan modulus yang dapat dipilih nilainya sebagaimana data nilainya ditampilkan pada Gambar 6. Sebagaimana ditampilkan pada Gambar 4 hasil pengujian yaitu pencarian data jurnal berdasarkan kemiripan judul terlebih dahulu sebelum menampilkan atau mencoba mengechek kemiripan abstrak pada jurnal.

Pengujian kemiripan diawali dengan menginput jumlah kata kunci, minimal 2 katakunci untuk mengkerucutkan

jumlah data yang ditampilkan [15]. Semakin banyak kata kunci semakin spesifik pencarian judul jurnal dan abstrak yang dicari untuk di cek kemiripannya sebagaimana ditunjukan pada Gambar 4.. Kemudian dapat dibatsijuga atau limit jumlah data yang diinput midal berdasarkan tahun. Berdasarkan refernsi kebaharuan, jurnal yang digunakan sebagai referensi baik atau idealnya adalah 5 tahun terakhir sehingga terupdate.



Gambar 4. Interface inputan kata kunci dan limit data

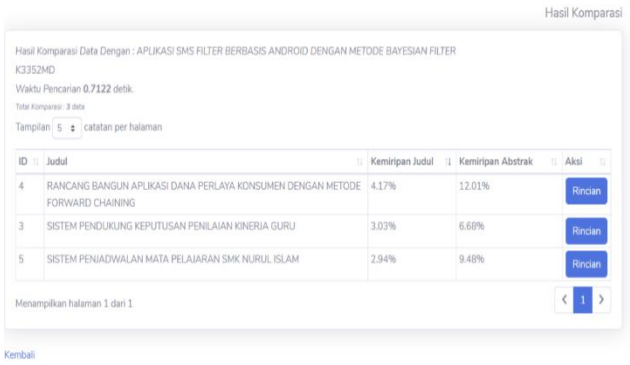
Pada aplikasi untuk nilai k-gram, modulus, basis dapat dipilih sesuai kebutuhan pengujian berdasarkan nilai n atau jumlah dokumen yang diuji. Dimana pada sebelah kanan apliksi terdapat *button* komparasi untuk menampilkan hasil perbandingannya ditunjukan pada Gambar 5.

Gambar 5. Hasil pengujian pencarian kemiripan judul jurnal

Gambar 6. Penggunaan nilai k-gram, basis, modulus dan dadu untuk berbagai jenis pengujian

Penggunaan nilai k-gram, basis, modulus dan dadu untuk berbagai jenis pengujian [11] pada Gambar 4 menujukan beberapa hasil yang berbeda saat pengujian, sehingga pada saat memilih model perlu diperhatikan mana penyumbang terbesar pada kenaikan akurasi algoritma. Salah satu yang mendasari Algoritma *Rabin-Karp* efisien, dalam menentukan *hash value*-nya[12].

Selanjutnya setelah diklik *button* komparasi maka tampilan pada Gambar 5 menunjukan nilai kemiripan paling besar terdapat pada ID jurnal 3, 4 dan 5 yaitu nilai terbesar kemiripan abstrak pada Jurnal 4 yaitu 12.01%, Jurnal 5 sebesar 9.48% dan Jurnal 3 sebesar 6.68%. Komparasi tersebut menggunakan nilai k-gram =7, modulo= 5007, dadu = 2 dan basis 7.



Gambar 8. Hasil pengujian k=5 perbandingan atau komparasi

Gambar 7 menunjukan nilai kemiripan paling besar terdapat pada ID jurnal 3, 4 dan 5 yaitu nilai terbesar kemiripan abstrak pada Jurnal 4 yaitu 12.01%, Jurnal 5 sebesar 9.48% dan Jurnal 3 sebesar 6.68%.

Gambar 8 menunjukan nilai kemiripan paling besar terdapat pada ID jurnal 4, 5 dan 3 yaitu nilai terbesar kemiripan abstrak 49.93% dan kemiripan judul 14.00%, Jurnal 5 sebesar 46.93% kemiripan abstrak dan 5.71% kemiripan jurnal, dan Jurnal 3 sebesar 40.91% pada kemiripan abstrak dan kemiripan judul 00.00 %.

Gambar 8. Hasil pengujian dengan k=7

Sedangkan pengujian yang menunjukan presentasi implementsi dialam kerja Rabin Karp yaitu modulo, basis dan k-gra, ditunjukan pada Tabel 1 dan Tabel 2. Berdasarkan hasil pengujian untuk warna merah yang

banyak mempengaruhi nilai k-gran, biru modulo dan hijau basis.

Tabel I. Hasil Pengujian 1

No Dokumen		Kemiripan	Kemiripan
No 1	No 2	1.96%	9.87%
No 1	No 3	0.00%	13.11%
No 1	No 4	6.26%	17.46%
No 1	No 5	2.94%	14.56%
No 1	No 6	4.88%	13.60%
No 1	No 7	5.66%	11.95%

Pengaruh penggunaan k-gram dan rolling hash terhadap nilai persentase suatu kemiripan dokumen menujukan tingkat yang tinggi pada algoritma Rabin Karp. Tingginya nilai k-gram yang tinggi maka semakin rendah nilai persentase dokumen, penggunaanya disarankan untuk kalimat yang panjang. Sedangkan basis dan modulus tidak terlalu signifikan terhadap persentase nilai kemiripan dokumen, namun nilai basis dan modulus yang tepat pada dokumen dapat membantu menurunkan atau meningkatkan nilai persentase dokumen

Tabel II. Hasil Pengujian 2

No Dokumen		Kemiripan	Kemiripan
No 1	No 2	0.00%	5.50%
No 1	No 3	0.00%	8.98%
No 1	No 4	4.17%	16.93%
No 1	No 5	0.00%	9.76%
No 1	No 6	4.88%	9.21%
No 1	No 7	3.77%	9.00%

IV. KESIMPULAN

Hasil pengujian pada penelitian ini dapat disimpulkan bahwa nilai *k-Grams* dan *rolling hash* perperan penting terhadap presentase tingkat similarity dokumen pada algoritma Rabin Karp, Panjang karakter < n disarankan untuk menggunakan nilai kgram antara 2 -3 sedangakan Panjang karater > n disarankan menggunakan Panjang karakter 4-7, hal tersebut menunjukan jika pengujian ingin lebih spesifik pada pengechekan kemiripan maka nilai k > n dan sebaliknya.

Hasil pengujian menunjukan k=7 menunjukan kemiripan lebih tinggi dibanding k=5 karena penelitian ini menggunakan panjang karakter jurnal > n dengan nilai kemiripan tertinggi yaitu k=7 pada dokumen jurnal 4 yaitu kemiripan abstrak 49.93% dan kemiripan judul 14.00% sedangkan untuk k=5 yaitu 12.01% kemiripan abstrak dan 4.17% kemiripan judul.

UCAPAN TERIMA KASIH

Tim penulis mengucapkan terima kasih kepada LPPM di Universitas Siliwangi yang berperan dalam pendanaan penelitian ini serta semua pihak yang terlibat dalam penyusunan jurnal.

DAFTAR PUSTAKA

[1] W. A. S. S L B Ginting, Y R Ginting, Sutomo, "Aplikasi Deteksi Kemiripan Kata Menggunakan Algoritma Rabin-Karp," *J. Teknol. dan Inf.*, vol. 12, no. 2, pp. 162–175, 2022, doi: 10.34010/jati.v12i2.

[2] T. Xplore, "Deteksi Plagiarisme Abstrak Skripsi dengan Menggunakan Algoritma Rabin Karp (Studi Kasus : Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang)," vol. 6, no. 2, pp. 75–81, 2021.

[3] A. Sunyoto and T. Informatika, "Implementasi Algoritma Rabin Karp untuk Pendeteksian Plagiat Dokumen Teks Menggunakan Konsep Similarity," vol. 2, no. 7, pp. 23–28, 2013.

[4] I. Widaningrum, D. Mustikasari, R. Arifin, and E. Dyah Cahyani, "Analisa Penggunaan K-Gram pada Karakter, Kata dan Kalimat untuk Mendeteksi Kesamaan Dokumen," *Pros. Semin. Nas. Teknoka*, vol. 5, no. 2502, pp. 59–64, 2020, doi: 10.22236/teknoka.v5i.333.

[5] F. Teknik, T. Informatika, U. Pamulang, T. Documents, and R. Algorithm, "Implementasi Algoritma Rabin-Karp Untuk Pendeteksian Plagiarisme Pada File Dokumen Berupa Text Berbasis Web," vol. 3, no. 3, pp. 150–154, 2022, doi: 10.47065/josh.v3i3.1404.

[6] R. Apriani *et al.*, "Analisis Sentimen dengan Naïve Bayes Terhadap Komentar Aplikasi Tokopedia," *J. Rekayasa Teknol. Nusa Putra*, vol. 6, no. 1, pp. 54–62, 2019, [Online]. Available: https://rekayasa.nusaputra.ac.id/article/view/86

[7] A. Filcha and M. Hayaty, "Implementasi Algoritma Rabin-Karp untuk Pendeteksi Plagiarisme pada Dokumen Tugas Mahasiswa," *JUITA J. Inform.*, vol. 7, no. 1, p. 25, 2019, doi: 10.30595/juita.v7i1.4063.

[8] A. Pratama Putra, Y. Pratama, E. Kharisma Krisnadi, I. Purnamasari, and D. Dwi Saputra, "Text Mining untuk Sentimen Analisis dengan Metode Naïve Bayes, SMOTE, N-Gram dan AdaBoost Pada Twitter CommuterLine," *J. Sains Komput. Inform. (J-SAKTI*, vol. 6, no. 2, pp. 961–973, 2022.

[9] H. Setiawan, E. Utami, and S. Sudarmawan, "Analisis Sentimen Twitter Kuliah Online Pasca

Covid-19 Menggunakan Algoritma Support Vector Machine dan Naive Bayes," *J. Komtika (Komputasi dan Inform.*, vol. 5, no. 1, pp. 43–51, 2021, doi: 10.31603/komtika.v5i1.5189.

[10] D. A. Putra and H. Sujaini, "Implementasi Algoritma Rabin-Karp untuk Membantu Pendeteksian Plagiat pada Karya Ilmiah," *J. Sist. dan Teknol. Inf.*, vol. 4, no. 1, pp. 66–74, 2015, [Online]. Available: http://jurnal.untan.ac.id/index.php/justin/article/view/12411

[11] D. N. Sari and D. P. Utomo, "Implementasi Algoritma Rabin-Karp Pada Pencarian Quotes Tokoh Terkenal," *Pelita Inform. Inf. dan Inform.*, vol. 9, no. 1, pp. 43–55, 2020.

[12] T. Jaringan, S. Bahri, and R. Wajhillah, "InfoTekJar : Jurnal Nasional Informatika dan Optimalisasi Algoritma Rabin Karp menggunakan TF-IDF Dalam Pencocokan Text Pada Penilaian Ujian Essay Otomatis," vol. 2, pp. 4–7, 2020.

[13] T. Tawang, I. Billhaqqi, and G. W. Wicaksono, "ANALISIS PERBANDINGAN ALGORITMA RABIN - KARP DAN WINNOWING DALAM PENILAIAN JAWABAN," pp. 269–276, 2020.

[14] T. Hidayat and M. Muttaqin, "Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," *J. Tek. Inform. UNIS JUTIS*, vol. 6, no. 1, 2018.

[15] A. Nisa, E. Darwiyanto, and I. Asror, "Analisis Sentimen Menggunakan Naive Bayes Classifier dengan Chi-Square Feature Selection Terhadap Penyedia Layanan Telekomunikasi," *e-Proceeding Eng.* , vol. 6, no. 2, p. 8650, 2019.