



Artikel

# IMPLEMENTASI ALGORITMA *RABIN-KARP* DAN *COSINE SIMILARITY* UNTUK PENDETEKSI PLAGIARISME PADA DOKUMEN

Eric Siswanto<sup>1</sup>, Yo Ceng Giap<sup>2</sup>*1,2 Universitas Buddhi Dharma, Teknik Informatika, Banten, Indonesia***SUBMISSION TRACK**

Received: March 01, 2020

Final Revision: May 03, 2020

Available Online: May 15, 2020

**KEYWORD**Algoritma *Rabin-karp*, Metode *Cosine Similarity*, Plagiarisme.**KORESPONDENSI**

Phone: 0819088804354

E-mail: [neolucky@yahoo.com](mailto:neolucky@yahoo.com)E-mail: [cenggiap@ubd.ac.id](mailto:cenggiap@ubd.ac.id)**A B S T R A K**

Pada era saat ini perkembangan teknologi informasi berkembang dengan sangat pesat dalam membantu banyak masalah pekerjaan. Khususnya pada kalangan mahasiswa untuk membantu dalam hal pembuatan tugas nya, Namun hal ini menyebabkan juga penyalahgunaan oleh mahasiswa yang semulanya agar dapat lebih mudah mencari referensi dengan mudah namun sekarang di gunakan sebagai alat untuk plagiarisme. Untuk itulah penulis membuat aplikasi yang dapat mendeteksi adanya plagiarisme pada dokumen yang bertujuan untuk mengetahui apakah dokumen tersebut di hasilkan dari melalui tindakan plagiarisme atau tidak. Metode yang di gunakan adalah Algoritma Rabin-Karp untuk mendeteksi kemiripan dokumen atau menganalisis kemiripan teks skripsi seperti pencocokan substring, kesamaan kata kunci pada dokumen dan metode cosine similarity untuk menghasilkan nilai similarity pada aplikasi pendeteksi plagiat ini. Aplikasi menggunakan bahasa pemrograman php, aplikasi notepad++ dan xampp. Pengujian sistem di lakukan dengan menggunakan metode white box dengan menganalisa alur aplikasi dan metode black box untuk menguji semua tombol pada aplikasi serta kuisoner untuk mengetahui pendapat dari para pengguna pada aplikasi ini. Aplikasi ini membandingkan 2 dokumen yang di masukan oleh pengguna untuk di uji persamaannya, jika nilai persamaannya melebihi 50 maka akan di tampilkan di halaman hasil perbandingan Berdasarkan hasil penelitian dapat di simpulkan bahwa dengan adanya aplikasi ini berhasil mencegah tindakan plagiarisme

**PENGANTAR**

Di era saat ini perkembangan teknologi informasi berkembang sangat pesat dalam membantu banyak masalah pekerjaan dengan cepat, akurat dan efisien. Perkembangan teknologi juga memudahkan kita untuk

bertukar dan mendapatkan informasi, terutama di antara siswa untuk membantu dalam membuat tugas mereka, tetapi ini juga menyebabkan penyalahgunaan oleh siswa yang awalnya digunakan agar lebih mudah menemukan referensi lebih mudah tetapi

sekarang di gunakan sebagai alat untuk duplikasi atau plagiarisme. Menurut Sukaesih [1], plagiarisme yaitu mengutip kata-kata atau kalimat orang lain tanpa menggunakan tanda kutip dan tanpa menyebutkan identitas sumbernya, menggunakan gagasan, pandangan atau teori orang lain tanpa menyebutkan identitas sumbernya, menggunakan fakta (data, informasi) milik orang lain tanpa menyebutkan identitas sumbernya, mengakui tulisan orang lain sebagai tulisan sendiri, melakukan parafrase (mengubah kalimat orang lain ke dalam susunan kalimat sendiri tanpa mengubah idenya) tanpa menyebutkan identitas sumbernya, dan menyerahkan suatu karya ilmiah yang dihasilkan dan atau telah dipublikasikan oleh pihak lain seolah-olah sebagai karya sendiri. Penyebab maraknya plagiarisme adalah sikap mental dan budaya para peneliti Indonesia yang ingin memperoleh sesuatu dengan mudah dan tidak terbiasa menghargai hasil karya orang lain. Hingga kini masyarakat Indonesia belum memiliki budaya untuk menghargai ilmu pengetahuan dan hak intelektual. Hal ini ditambah sistem pendidikan yang sejak awal tidak mendidik orang menjadi kreatif. Ketika di kelas para siswa umumnya kurang memperoleh kesempatan berbicara bebas. Hal tersebut mematikan kreativitas Sehingga membuat orang menjadi malas, sehingga sejak bangku Sekolah Dasar budaya mencontek menjadi suatu hal yang lumrah dilakukan oleh para siswa. Menurut Sukaesih [1], Plagiarisme dianggap merugikan, tidak hanya bagi penulis yang karyanya disalin, tetapi juga menyebabkan kurangnya kreativitas dan menyebabkan pembentukan mental yang buruk bagi mereka yang melakukannya. Plagiarisme menyebabkan kurangnya kreativitas pada pelakunya dan menunjukkan kemalasan pelakunya dalam menemukan ide-ide baru dalam penulisan karya ilmiahnya. Untuk alasan ini penulis membuat aplikasi yang dapat mendeteksi keberadaan plagiarisme dalam dokumen yang bertujuan untuk mengetahui apakah dokumen tersebut dihasilkan melalui plagiarisme atau tidak. Dalam hal ini penulis menggunakan

Algoritma Rabin-Karp dan metode kesamaan cosinus dalam aplikasi pendeteksi plagiarisme ini.

## I. METODE

### 1.1. Algoritma *Rabin-Karp*

Menurut Purba dan Sitorus [4], Algoritma *Rabin-Karp* adalah algoritma pencocokan *string* yang menggunakan *fungsi hash* sebagai pembandingan antara *string* yang dicari (m) dengan *substring* pada teks (n). Apabila *hash value* keduanya sama maka akan dilakukan perbandingan sekali lagi terhadap karakter-karakternya. Apabila hasil keduanya tidak sama, maka *substring* akan bergeser ke kanan. Pergeseran dilakukan sebanyak (n-m) kali, Perhitungan nilai *hash* yang efisien pada saat pergeseran akan mempengaruhi performa dari algoritma ini.

Pada dasarnya, algoritma *Rabin-Karp* akan membandingkan nilai *hash* dari *string* masukan dan *substring* pada teks. Apabila sama, maka akan di lakukan perbandingan sekali lagi terhadap karakter-karakternya. Apabila tidak sama, maka *substring* akan bergeser ke kanan. Kunci utama performa algoritma ini adalah perhitungan yang efisien terhadap nilai *hash substring* pada saat penggeseran dilakukan.

Menurut Herriyance, Handrizal dan Fadilla [2], Algoritma *Rabin-Karp* diciptakan oleh Michael O. Rabin dan Richard M. Karp pada tahun 1987 yang menggunakan fungsi *hashing* untuk menemukan *pattern* di dalam *string* teks. Charas and Lecroq, didalam bukunya menyatakan bahwa fungsi *hashing* menyediakan metode sederhana untuk menghindari perbandingan jumlah karakter yang kuadrat di dalam banyak kasus atau situasi. Dari pada melakukan pemeriksaan terhadap setiap posisi dari teks ketika terjadi pencocokan pola, akan lebih baik dan efisien melakukan pemeriksaan hanya jika teks yang sedang proses memiliki kemiripan seperti pada *pattern*. Untuk melakukan pengecekan kemiripan antara dua kata ini digunakan fungsi *hash*. Ramdhani, dalam penelitiannya menuliskan bahwa fungsi *hash* menyimpan bentuk *string* dalam bentuk lain yaitu

enumerasi sehingga suatu *string* tertentu akan memiliki nilai enumerasinya yang unik. Raharja meneliti tentang perancangan dan implementasi sistem penilaian jawaban esai otomatis menggunakan algoritma *Rabin-Karp*, dimana untuk mencari nilai *hash* digunakan persamaan :

$$H = C_1 * B^{(m-1)} + C_2 * B^{(m-2)} + \dots + C_{(m-1)} * B^m + C_m$$

Dimana:

H = Nilai Hash

C = ASCII karakter

B = Basis (bilangan prima)

m = Banyak karakter (panjang karakter)

Berdasarkan teori langkah-langkah algoritma *Rabin-Karp* diatas, maka disediakan contoh data yang akan diuji yaitu:

Kalimat 1 : Aplikasi *game binggo online* menggunakan algoritma *minimax*. Kalimat 2 : Smart kontrol temperatur menggunakan sensor LM35 berbasis mikrokontroler AT89S52.

- a. Menghilangkan tanda baca, menghapus spasi dan mengubah teks menjadi kata-kata tanpa huruf kapital.

Kalimat 1:

Aplikasigamebinggoonlinemenggunakan algoritmaminimax

Kalimat 2:

smartkontroltemperaturmenggunakan sensorlumberbasismikrokontrolerats

- b. Menentukan *k-gram*, basis bilangan dan membagi teks ke dalam *gram-gram* yang sudah ditentukan nilai *k-gram* nya.

*K-gram* = 5 (lima)

Basis bilangan = Desimal (10)

Bentuk *gram* kalimat 1 :

aplik|plika|likas|ikasi|kasig|lasiga|sigam|igame|gameb|amebi|mebin|ebing|bingg|inggo|nggoo|ggoon|goonl|oonli|onlin|nline|linem|ineme|nemen|emeng|mengg|enggun|gun|ggun|gunagunak|unaka|nakan|akana|kan

al|anal|nalgo|algor|lgori|gorit|oritm|ritma|itmam|tmami|mamin|amini|minim|inima|nimax|

Bentuk *gram* kalimat 2 :

smart|martk|artko|rtkon|tkont|kontr|ontro|ntrol|trolt|rolte|oltem|ltemp|tempe|emper|mpera|perat|eratu|ratur|aturm|turme|urmen|rmeng|mengg|enggu|nggun|gguna|gunak|unaka|nakan|akans|kanse|ansen|nsens|senso|ensor|nsorl|sorlm|orlmb|rlmbe|lamber|mberb|berba|erbas|rbasi|basis|asism|sismi|ismik|smikr|mikro|ikrok|kroko|rokon|okont|kontr|ontro|ntrol|troll|rolle|oller|llera|lerat|erats.

- c. Mencari nilai *hash* dengan fungsi *rolling hash* dari tiap *gram* yang terbentuk.

Nilai *hash* dari *k-gram* aplik :

$$\begin{aligned} \text{Aplik} &= \text{Ascii (a)} 97 * 10^5 - 1 + \text{Ascii (p)} 112 * 10^4 - 1 + \text{Ascii (l)} 108 * 10^3 - 1 \\ &+ \text{Ascii (i)} 105 * 10^2 - 1 + \text{Ascii (k)} 107 * 10^1 - 1 \\ &= 97 * 10^4 + 112 * 10^3 + 108 * 10^2 + 105 * 10^1 + 107 * 10^0 \\ &= 97 * 10000 + 112 * 10000 + 108 * 1000 + 105 * 100 + 107 * 10 \\ &= 970000 + 112000 + 10800 + 1050 + 107 \\ &= 1093957. \end{aligned}$$

## 1.2. Cosine Similarity

Menurut sugiyanto, Surarso dan Aris Sugiharto [3], *Cosine similarity* merupakan metode yang di gunakan untuk menghitung tingkat kesamaan (*similarity*) antar dua buah objek. Untuk tujuan klastering dokumen, fungsi yang baik adalah fungsi *cosine similarity* (Salton, 1989). Untuk notasi himpunan dapat di gunakan rumus.

$$\text{Similarity}(X, Y) = \frac{|X \cap Y|}{|X|^{\frac{1}{2}} \cdot |Y|^{\frac{1}{2}}}$$

Di mana  $|X \cap Y|$  adalah jumlah term yang ada pada dokumen X dan yang ada pada dokumen Y,  $|X|$  adalah jumlah term yang ada pada dokumen X dan  $|Y|$  adalah jumlah term yang ada pada dokumen Y.

$$Similarity(X, Y) = \frac{\sum_{i=1}^i x_i y_i}{\sqrt{\sum_{i=1}^i X_i^2 \cdot \sum_{i=1}^i Y_i^2}}$$

Di mana x dan y adalah dokumen yang berbeda, Xi adalah istilah i dalam dokumen x sedangkan Yi adalah istilah i yang ada di dokumen Y.

**II. HASIL**

**2.1. Hasil Perbandingan file dokumen dengan format Doc, Docx, Pdf dan Txt**

**Table 1. Tabel Produk Yang Sering Di Beli**

Dokumen 1	Dokumen 2	Waktu Proses	Hasil
Sistem Pakar Diagnosa Depresi Mahasiswa Akhir Dengan Metode	Pengembangan Media Pembelajaran Berbasis Augmented Reality	02,88 - 03,38 Detik	32%
Implementasi Global Positioning System GPS dan.pdf	Landasan Teori	03,32 - 03,46 Detik	33%
Perancangan Aplikasi Untuk Memeriksa	Latar Belakang Masalah doc	01,30 - 01,66 Detik	27 %

Keaslian Data			
---------------	--	--	--

**III. PEMBAHASAN**

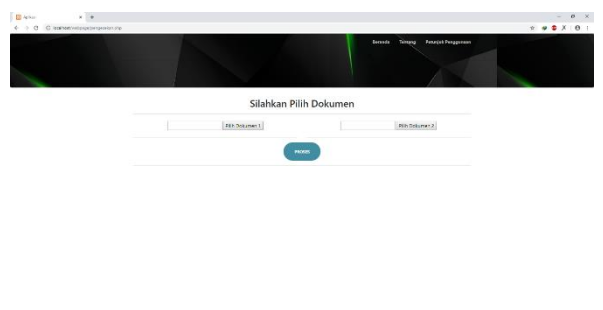
**3.1 Tampilan Program**

a. Halaman tampilan awal



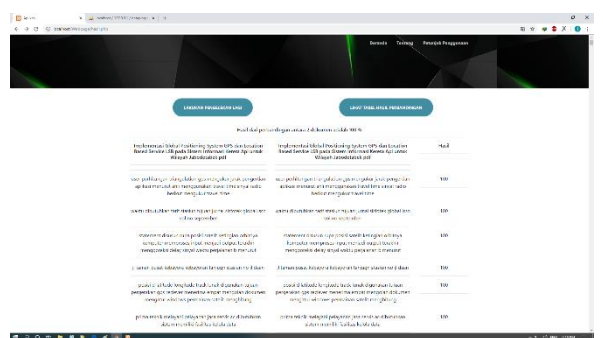
Gambar III.1 Halaman tampilan awal

b. Halaman memulai pengecekan



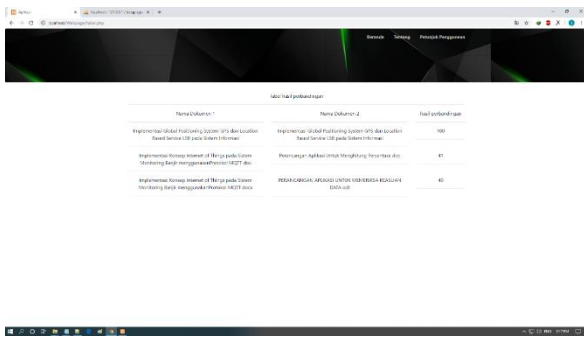
Gambar III.2 Halaman memulai pengecekan

c. Halaman hasil pengecekan



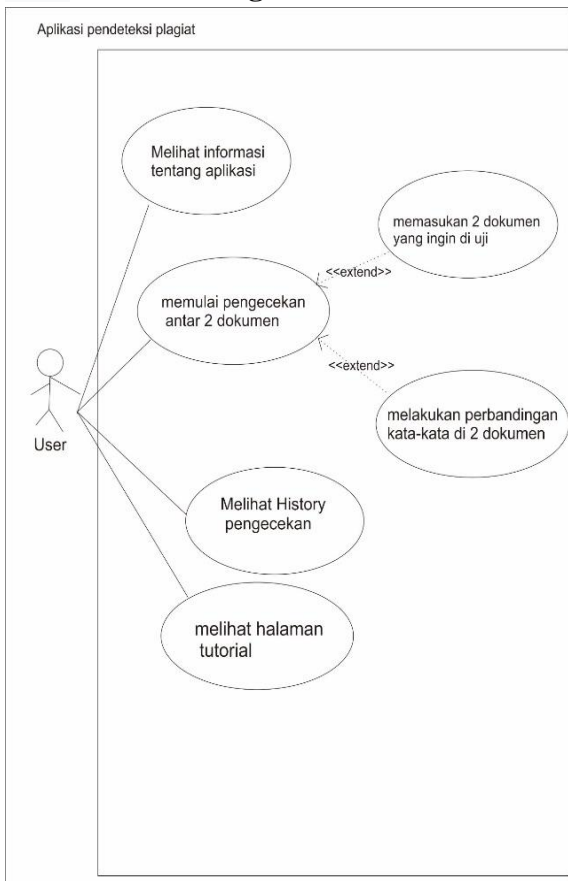
Gambar III.3 Halaman hasil pengecekan

d. Halaman Tabel Hasil Pengecekan



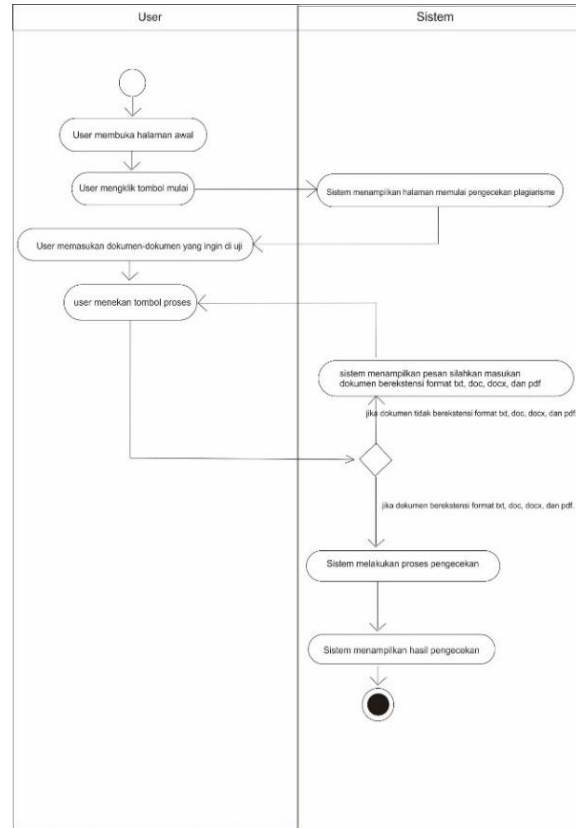
Gambar III.3 Halaman tabel hasil pengecekan

3.2 Use Case Diagram



Gambar III.4 Use Case Diagram

3.3 Activity Diagram



Gambar III.5 Activity Diagram

IV. KESIMPULAN DAN SARAN

4.1. Kesimpulan

Berdasarkan uji coba dan evaluasi yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut:

- Algoritma *Rabin-karp* dan metode *cosine similarity* dapat di implementasikan di aplikasi sesuai dengan *rule*.
- Pengguna dapat membandingkan dokumen dengan ekstensi *format .doc, .docx, .pdf, dan .txt*.
- Berdasarkan uji coba pada aplikasi dengan menggunakan algoritma *Rabin-karp* dan metode *Cosine Similarity* memproses perbandingan plagiarisme dengan jumlah karakter sebanyak 42432 memakan waktu sebanyak 12,49 - 16,17 Detik, waktu bertambah berdasarkan banyaknya jumlah karakter dalam dokumen.
- Berdasarkan kuesioner yang telah di bagikan dan di isi oleh para pengguna, dapat di simpulkan bahwa aplikasi ini

menghasilkan hasil perbandingan yang akurat, mudah di gunakan dan membantu menjaga dokumen dari tindakan plagiarisme.

#### **4.2. Saran**

Berdasarkan analisa yang telah di lakukan maka dapat di usulkan beberapa saran agar aplikasi pembelajaran ini lebih baik lagi, yaitu:

- a. Pengembangan aplikasi dapat di lakukan dengan menerapkan metode lain agar dapat di ketahui performa dan kekurangan dari setiap metode.
- b. Pengembangan aplikasi dapat di lakukan dengan membandingkan gambar juga di dalam dokumen.
- c. Pengembangan selanjutnya di harapkan dapat membandingkan dokumen yang ada di internet.

## REFERENSI

- [1] M.Si. Sukaesih. Dra. (2018). Permasalahan Plagiarisme Dalam Penelitian Kualitatif Di Indonesia. Jurnal Politikom Indonesiana, E-ISSN : 2528 – 2069.
- [2] Herryance Dan Handrizal Dan Fadilla, Siti Dara. (2017). Analisis Algoritma Rabin-Karp Pada Kamus Umum Berbasis Android. Jurnal Riset Sistem Informasi Dan Teknik Informatika (Jurasik), Volume (2) No. 1 ISSN: 2527-5771/EISSN: 2549-7839.
- [3] Sugiyanto. Surarso, Bayu dan Sugiharto Aris (2014), Analisa Performa Metode Cosine Dan Jaccard Pada Pengujian Kesamaan Dokumen, Jurnal Masyarakat Informatika, Volume 5, Nomor 10, ISSN 2086-4930.
- [4] Purba, Ledisma Juliana dan Sitorus, Lamhot. (2018). Perancangan Aplikasi Untuk Menghitung Persentase Kemiripan Proposal Dan Isi Skripsi Dengan Algoritma Rabin-Karp , Volume 03 Nomor 01 ISSN : 2548-1916.

## BIOGRAPHY

**Eric Siswanto, S.Kom**, dilahirkan di tangerang, 29 maret 1996, Sekolah Dasar Yunike Andreas, Kota Tangerang, Sekolah Menengah Pertama Mawar Saron, Kota Tangerang, Sekolah Menengah Kejuruan Buddhi, Kota Tangerang, Pendidikan S-1 (Teknik Informatika – Universitas Buddhi Dharma) lulus tahun 2020.

**Yo Ceng Giap, M.Kom., CPS**, Saat ini bekerja sebagai dosen Tetap pada Program Studi Teknik Informatika di Universitas Buddhi Dharma.