U-NET Jurnal Teknik Informatika
Vol. 7, No. 1, Februari 2023, pp. 1~12
e-ISSN: 2622-8238, p-ISSN: 2722-4287
DOI: https://doi.org/xxxxxxxxxxxxxxxxxx

Identifikasi Tingkat Kemiripan Dokumen Teks Menggunakan Fungsi Hash Pada Algoritma Winnowing dan Pattern Recognation Pada Algoritma Rafcliff/Obershelp

Abdul Halim Hasugian[1], Muhammad Siddik Hasibuan[2], Yusuf Karim Rambe[3]*
[123]Program Studi Ilmu Komputer, Universitas Islam Negeri Sumatera Utara, Medan, Indonesia
Email: abdulhalimhasugian@uinsu.ac.id[1], muhammadsiddik@uinsu.ac.id[2] , ykrambe@gmail.com[3]

Article Info

Article history:

Received 28 01 2023
Revised 31 01 2023
Accepted 03 02 2023

Keyword:

Plagiarisme
Dokumen
Winnowing
Ratcliff/Obershelp

Correspondence Author*:
Nama : Yusuf Karim Rambe
Email : ykrambe@gmail.com

Abstract

Perkembangan teknologi yang sudah sangat maju ini dalam bidang komputer, sekarang semua dokumen sudah berubah menjadi file softcopy yang mana bisa diakses lewat komputer ataupun lewat smartphone masing-masing. Sehingga banyak yang melakukan penjiplakan dan tidak dicantumkan sumbernya sehingga terjadi pelanggaran hak cipta. Plagiarisme dapat dipahami sebagai tindakan mengambil pernyataan atau mencuri ide orang lain, dalam hukum positif, hak paten dan kekayaan intelektual sudah ada diatur dalam undang-undang Oleh karena itu untuk memudahkan para guru dan dosen dalam bidang akademik, dibutuhkan pengidentifikasian pada dokumen dokumen tersebut agar diketahui apakah dokumen tersebut termasuk plagiarisme dengan mendeteksi kemiripan teks antar dokumen, maka keaslian dalam tiap dokumen atau karya tulis bisa tetap terjaga keasliannya. Pengidentifikasian akan dilakukan dengan algoritma Winnowing dan Ratcliff/Obershelp. Algoritma Winnowing adalah algoritma yang digunakan untuk melakukan proses pengecekan kesamaan kata (document fingerprinting) untuk mengidentifikasi penjiplakan. Ratcliff/Obershelp juga untuk mendeteksi adanya kemiripan namun dalam prinsip kerjanya dilakukan pengembalikan nilai yang dapat digunakan sebagai persentase dalam menunjukkan kesamaan dua string dengan meperhitungkan jumlah karakter yang terdapat pada kedua string tersebut. Kedua algoritma ini dapat digunakan dalam mengidentifikasi tingkat kemiripan dokumen teks.

1. PENDAHULUAN

Seiring dengan perkembangan teknologi yang sudah sangat maju ini sekarang semua file sudah berubah menjadi softcopy yang mana bisa diakses melalui gadget masing-masing. Namun entah dikarenakan kesibukan ataupun kelalaian dan mungkin karena terlalu teknologi terlalu praktis maka plagirisme inipun terjadi, banyak yang melakukan penjiplakan dan tidak dicantumkan sumbernya sehingga terjadi pelanggaran hak cipta. Plagiarisme adalah tindakan mengambil pernyataan atau ide orang lain dan mengklaimnya sebagai milik Anda sendiri. Bahkan jika ide atau pernyataan itu milik orang lain, itu telah diambil dan sumbernya tidak disebutkan [1], sedangkan dalam hukum islam itu sendiri, hak kekayaan intelektual termasuk kedalam (huquq maliyyah) yang mendapatkan perlindungan hukum (mashun)[2]. Oleh karena itu, dibutuhkan adanya pendeteksian dalam dokumen atau karya tulis agar diketahui apakah termasuk plagiat atau tidak, dengan mendeteksi kemiripan antar dokumen maka keaslian dalam tiap dokumen atau karya tulis bisa tetap terjaga keasliannya, karena saat ini mudah sekali dilakukan plagiarism.

Plagiarism sendiri punya tingkatan agar bisa dianggap sebuah plagiarisme, berikut adalah tingkat agar dinyatakan sebuah plagiarism;

a. 0% dianggap tidak ada kesamaan antara dokumen uji dengan dokumen pembanding.
b. 15% dianggap memiliki beberapa kesamaan antara dokumen teks dan dokumen pembanding tetapi tidak mengklaim sebagai plagiarisme.

c. 15-50% dianggap memiliki tingkat kemiripan sedang pada dokumen yang di periksa dan dibandingkan disebut plagiarism sedang.
d. 50% dianggap telah menemukan plagiarisme dalam dokumen yang diperiksa dan dibandingkan.
e. 100% dianggap bahwa kedua dokumen memiliki nilai kesamaan yang mutlak plagiarism.

Beberapa penelitian sebelumnya telah membuat penelitian pendeteksian *plagiarism* dengan metode Winnowing[3] yang mana algoritma ini adalah algoritma yang digunakan untuk melakukan pemeriksaan kemiripan kata (document fingerprinting) untuk mengidentifikasi plagiarisme. Algoritma yang digunakan untuk mencari hash pada Winnowing adalah Rolling Hash. Nilai hash adalah nilai numerik yang terbentuk dari perhitungan ASCII dari setiap karakter atau angka yang ada. Dan juga dengan menggunakan algoritma ratcliff/Obershelp[4]. Yang mana algoritma ini juga untuk mendeteksi adanya kemiripan namun dalam prinsip kerjanya dilakukan pengembalikan nilai yang dapat digunakan sebagai persentase dalam menunjukkan kesamaan dua string dengan memerhitungkan jumlah karakter yang terdapat pada kedua string tersebut.
Berdasarkan kedua terdahulu yang telah dicantumkan di atas, penulis mengambil kesimpulan bahwa algoritma Winnowing dan Ratcliff/Obershelp cocok untuk dikombinasikan karena kedua algoritma ini saling melengkapi satu sama lain yang mana Winnowing mengidentifikasi kemiripan dokumen teks dengan fungsi hash yang diubah menjadi fingerprint, sedangkan algoritma Ratcliff/Obershelp mengidentifikasi kemiripan dokumen teks berdasarkan keseluruhan teks pada dokumen yang akan diuji.

2. METODOLOGI PENELITIAN

Metode pendeteksian plagiarisme dapat dibagi menjadi tig, yaitu perbangingan teks lengkap, dokument fingerprint dan kesamaan kata kunci. Pada algoritma Winnowing termasuk pada metode fingerprint dan sedangkan pada algoritma Ratcliff/Obershelp termasuk keadalam metode terbandingan teks lengkap dan kesamaan kata kunci. Sedangkan untuk proses algoritma dapat dilihat pada subbab di bawah ini:

2.1 Preprocessing

Preprocessing adalah proses yang awalnya menghilangkan bagian-bagian yang tidak perlu dari pemrosesan data. Pada titik ini, beberapa proses dilakukan pada algoritma Winnowing . Berikut adalah macam-macam dari metode prepocessing pada algoritma Winnowing [5], yaitu:

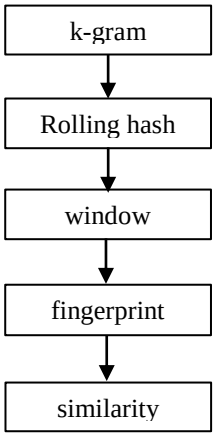
a. Case folding, proses dimana dilakukan pengubahan semua karakter huruf kapital mencadi huruf kecil atau lowercase.
b. Filtering atau stopword removal, adalah proses penghapusan kata-kata yang tidak relevan dalam teks nantinya.
c. Tokenizing, memisahkan kata berdasarkan dari susunan kata dan biasanya dipisahkan oleh karakter whitespace.

Sedangkan pada algoritma Ratfliff/Obershelp preprocessing nya agak sedikit berbeda. Berikut adalah macam-macam dari metode prepocessing pada algoritma Ratfliff/Obershelp, yaitu:

a. Case folding, proses dimana dilakukan pengubahan semua karakter huruf kapital mencadi huruf kecil atau lowercase.
b. Cleansing, dokumen akan di bersihkan dari tanda baca dan simbol-simbol yang tidak di perlukan.
c. Filtering atau stopword removal, proses penghapusan kata-kata yang tidak relevan dalam teks nantinya.
d. Penghapusan spasi, dilakukan penghapusan spasi, agar menjadi satu string.

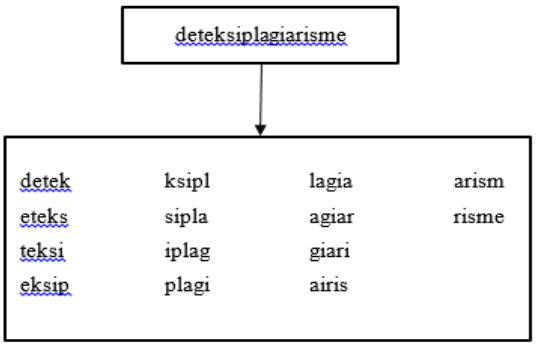
2.2 Algoritma Winnowing

Algoritma Winnowing adalah metode untuk meningkatkan efisiensi pencocokan dokumen sidik jari. Input pada algoritma ini adalah dokumen teks yang diproses untuk menghasilkan output berupa kumpulan hash, nilai hash berupa nilai numerik yang terbentuk dari perhitungan ASCII setiap karakter. Kumpulan hash ini kemudian disebut sidik jari. Sidik jari ini akan digunakan untuk membandingkan kesamaan antara dokumen teks. Langkah-langkah yang diterapkan pada algoritma Winnowing secara singkat dapat di rangkup menjadi seperti di bawah ini;



Gambar 1. Skema algoritma Winnowing

Langkah awal dalam penerapan algoritma Winnowing adalah melakukan lowercase pada setiap karakter dan membuang karakter-karakter dari dokumen yang tidak relevan misalnya tanda baca, spasi, dan simbol lain [6].
Langkah kedua, isi dokumen yang telah dilakukan permbersihan pada langkah preprosessing, dilanjutkan dengan membentuk rangkaian gram. Dimana nilai k = 5. Seperti terlihat pada gambar berikut:



Gambar 2. Progses k-gram

Langkah ketiga dilakukan proses hash untuk menghasilkan nilai hash dari setiap gram yang terbentuk. Dari proses tersebut didapatkan nilai hash dari setiap gram sebagai barikut.

12281, 12658, 13536, 12532, 13161, 13579, 12895, 13275, 12706, 11988, 12498, 12580, 12334, 13532.

Langkah keempat membentuk window. Proses pembentukan window sama seperti proses k-gram dari nilai-nilai hash yang dihasilkan dengan besar window w = 4:

{12281, 12658, 13536, 12532}, {12658, 13536, 12532, 13161}

{13536, 12532, 13161, 13579}, {12532, 13161, 13579, 12895}

{13161, 13579, 12895, 13275}, {13579, 12895, 13275, 12706}

{12895, 13275, 12706, 11988}, {13275, 12706, 11988, 12498}

{12706, 11988, 12498, 12580}, {11988, 12498, 12580, 12334}

{12498, 12580, 12334, 13532}

Langkah kelima adalah memilih hash terkecil dari setiap window untuk dijadikan fingerprint dokumen tersebut, dari window diatas didapatkan fingerprints:

12281, 12532, 12895, 12706, 11988, 12334

Setelah proses ini, dilajutkan dengan menghitung tingkat kemiripan menggunakan *coeficient jaccard similarity*. Untuk lebih memahami masing-masing proses proses yang dilakukan bisa di lihat pada point berikut.

2.2.1 K-Gram

K-gram adalah urutan token dengan memecah teks menjadi string yang dimulai pada beberapa posisi dalam teks dengan panjang k. Dalam konteks linguistik komputasi, token ini dapat berupa kata-kata, meskipun dapat berupa karakter atau subset karakter[7]. Metode k-gram terdiri dari membuat substring dari k karakter dari sebuah string. sebagai contoh; misalkan sebuah nama "YUSUF KARIM RAMBE", dari 3 kata tersebut dipotong string k, misalkan nilai k=3, dari kalimat tadi maka dapat membentuk hasil sebagai berikut:

"YUS", "UFK", "ARI", "MRA", "MBE"

Metode k-gram itu sendiri mempunyai peran yang cukup penting, karena dari sinilah nantinya proses fingerprint akan dibentuk. Dengan kata lain, metode ini memiliki pengarah yang besar untuk hasil yang dikeluarkan nantinya. Pengaruh dari nilai K yang digunakan pada metode K-Gram yaitu semakin kecil nilai k maka akan semakin besar pula perssentase yang dihasilkan nantinya.

2.2.2 Hash

Hash atau roll hash adalah fungsi yang menerima atau mengubah beberapa string input dan mengubahnya menjadi string output dengan panjang tetap. Fungsi hash terdiri dari dua elemen, fungsi hash dan nilai hash. Hash yang digunakan pada algoritma Winnowing adalah fungsi hash yang akan mengubah setiap karakter dalam string string menjadi kode ASCII [8], dan rumusnya seperti berikut:

$H(c_1 \dots c_k) = c_1 * b^{(k-1)} + c_2\ b^{(k-2)} * \dots \dots + c_{(k-1)} * b^{(k)} + c_k$ (2.1)

Keterangan:

H = nilai hash

c = nilai ASCII karakter

b = basis (bilangan prima)

k = banyak karakter

Untuk menghemat waktu saat menghitung hash k-gram nanti, persamaan roll hash (2.2) dapat digunakan sebagai berikut:

$H(c_2 \dots c_{k+1}) = H\ (c_1 \dots . c_k)$ - $c_1 * b^{(k-1)} * b + c_{(k)}$ (2.2)

Oleh karena itu, tidak perlu melakukan iterasi dari indeks pertama ke indeks akhir untuk menghitung hash untuk k-gram ke-2 dan k-gram terakhir.

2.2.3 Window

Proses Winnowing membutuhkan parameter w-gram dimana hash yang dihasilkan di sub-rantai untuk menghasilkan urutan hash W. Dari urutan hash ini, hash terkecil dipilih.dan jika kemudian 2 atau lebih dari nilai yang sama, hash yang lebih kecil akan dipilih di paling kanan[9].

2.2.4 Fingerprint

Prinsip pengoperasian metode ini adalah menggunakan hashing, hashing adalah fungsi yang mengubah setiap string menjadi angka dan menyimpannya dalam diagram atau grafik. Pilihan dalam metode sidik jari adalah memilih nilai hash terkecil di setiap jendela dokumen [10]. Algoritme penampi sendiri tidak hanya mengambil nilai sidik jari tetapi juga lokasi sidik jari dalam dokumen.

2.2.5 Jaccard's Similarity Coeficient

Jaccard's Similarity Coeficient adalah persamaan yang digunakan untuk menghitung atau mengukur kesamaan atau ketidaksamaan teks dalam suatu dokumen[11]. Ukur faktor kesamaan Jaccard dari satu dokumen panggilan, maka hasil kedua dokumen akan dibagi dua dan dikalikan dengan 100%:

Similaritas $(d_i, d_j) = \frac{|w(d_i) \cap w(d_j)|}{|w(d_i) \cup w(d_j)|}$ x 100% (2.3)

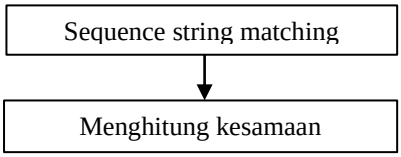
Keterangan:

1. $|w(d_i) \cap w(d_j)|$: irisan fingerprint dokumen dengan dokumen pembanding,

2. $|w(d_i) \cup w(d_j)|$: gabungan fingerprint dokumen dan dokumen pembanding,
3. Similaritas (d_i, d_j) : hasil bagi irisan dan gabungan fingerprint yang dikalikan 100%.

2.3 Algoritma Ratcliff/Obershelp

Algoritma Ratcliff/Oberhelp menggunakan proses yang sama untuk memutuskan seberapa mirip dua pola satu dimensi. Karena string teks merupakan satu dimensi[12]. Algoritma ini mengembalikan nilai yang dapat digunakan sebagai faktor kepercayaan atau persentase menunjukkan kesamaan dua string.



Gambar 3. Skema algoritma Ratcliff/Obershelp

Pada tahapannya setelah dilakukan preprocessing yang telah dibahas sebelumnya kemudian tahapan yang dilakukan selanjutnya dalam algoritma Ratcliff/Obershelp adalah sebagai berikut:

1. Sequence (String) Matching

a. Menghitung banyaknya karakter.

Pencarian nilai |S1| dan |S2| dimana |S1| adalah banyaknya karakter yang dimiliki oleh string1 sedangkan |S2| adalah banyaknya string yang dimiliki oleh string2, bisa dilihat contohnnya sebagai berikut;
String 1

Minumkopimenghilangkankantuk => |S1| = 28

String 2
Minumkopidinginmenghilangkankantukdanhaus => |S2| = 41

b. Pencarian subsequence (substring)
Menghitung pencarian subsequence (substring terpanjang dinamakan anchor) dan mencari substring lainnya dari kedua string.
String 1
Minumkopi|menghilangkankantuk |
String 2
Minumkopidingin|menghilangkankantuk|danhaus
Anchor : menghilangkankantuk => |Km| = 19

Menggabungkan substring yang telah ditemukan dari kedua string, maka tolal karakter dari substring tersebut dijumlahkan, maka

Km = 19 +9 = 28

2. Menghitung kesamaan.

Setelah semua tahapan itu dilakukan, dan substring sudah ditemukan, maka tehap selanjutnya adalah penilanan algoritma Ratcliff/Obershelp dilakukan dengan rumus perhitungan sebagai berikut:

$$\text{Dro} = \frac{2\ x\ Km}{|s1|+|s2|} \text{ x } 100\% \quad \text{.....(2.4)}$$

Keterangan:

Km = jumlah karakter yang sama

S1| = panjang dari string 1

|S2| = panjang dari string 2

Sebagai contoh, jika ingin mencari kesamaan kedua string dari PLAGIARISM dan PLAGIAT

a. Panjang dari string S_1 : $|s_1|$ = 10
Panjang dari string S_2 : $|s_2|$ = 7

b. Substring yang terpanjang yang sama adalah PLAGIA, maka PLAGIA merupakan sebuah anchor maka Km=|PLAGIA| = 6

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
S1	P	L	A	G	I	A	R	I	S	M
S2	P	L	A	G	I	A	T			

Tabel 1. Contoh pengambilan substring

c. Jika misalkan nantinya masih terdapat anchor lain kita tinggal menambahkan jumlah anchor itu pada Km, pada kasus ini kita mendapati nilai dari Km adalah 6
d. Disebelah Km terdapat karakter RISM dan T, karena tidak ditemukan kecocokan maka mereka tidak cocok, dan nilai Km tetap menjadi 6, setelah kita memiliki semua data yang di perlukan maka kita mulai perhitungannya

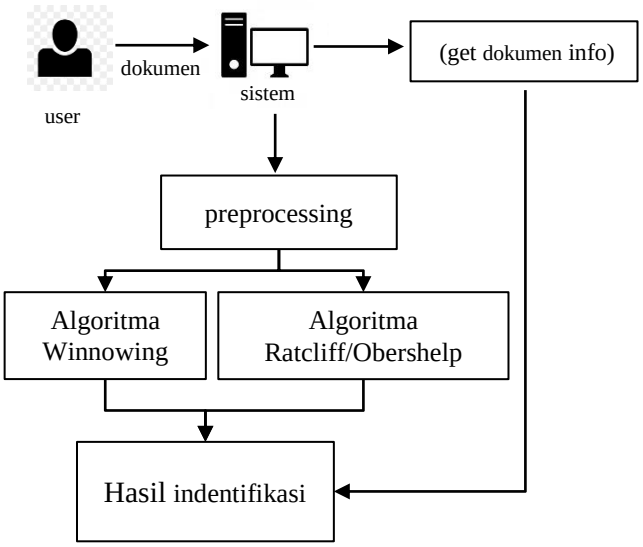
Penilaian Ratcliff/Obershelp dari string PLAGIARISM dan PLAGIAT adalah:

$$Dro = \frac{2\ x\ 6}{10+6} \text{ x } 100\% \ = \ \frac{12}{16} \text{ x } 100\% \ = \ 75\%$$

Jadi dari kedua string PLAGIARISM dan PLAGIAT memiliki kesamaan 75% yang bisa dikatakan sama, Algoritma Ratcliff/Obershelp memiliki kelebihan dan kekurangan yang hampir sama dengan algoritma Rabin-Karp yaitu proses komputasi yang lebih mudah[13], yang dapat digunakan dalam hal pencarian string dengan pola yang lebih panjang dan langkah pemrosesan yang lebih pendek. Sedangkan kelemahannya adalah tidak dapat menentukan kesamaan makna sinonim, dan keakuratan pendeteksiannya sangat dipengaruhi oleh posisi kalimat[14].

2.4 Perancangan sistem

Perancangan sistem yang akan digunakan untuk membuat aplikasi deteksi plagiarisme menggunakan algoritma Winnowing dan Ratfliff/Obershelp adalah sebagai berikut;



Gambar 4. Tahapan algoritma Winnowing dan Ratcliff/Obershelp

Sistem dimulai dengan menginput dokumen yang akan di identifikasi. Ketika dokumen diinput maka system akan membaca dokumen tersebut dan akan dimulai proses preprocessing yang dimulai dengan case folding, pada tahap ini teks dokumen yang di input akan di ubah setiap hurufnya menjadi huruf kecil (lowercase), setelah itu ke proses filtering, ditahap ini teks dokumen akan dilakukan penghapusan kata-kata yang tidak relevan, menghapus tanda baca dan symbol yang tidak di perlukan, menghapus karakter whitespace, yang pada akhirnya semua teks dokumen akan diubah menjadi sebuah string atau kalimat yang panjang. Lalu dilanjutkan dengan proses tokenizing, yang mana pada tahap ini hasil string panjang yang di dapat dari

proses filtering akan di potong-potong menjadi sekumpulan token atau kata berdasarkan panjang k-gram. Dan dilanjutkan dengan mengkonversi hasil dari k-gram menjadi rolling hash dan memisahkannya berdasarkan window menentukan fingerprint dari masing-masing teks dokumen dan pada akhirnya menentukan tingkat similarity dengan jaccard similarity coefficient dan menampilkan output dari proses tersebu. Lalu dilanjutkan dengan algoritma Ratcliff/Obershelp dengan mengambil data preprocessing yang sebelumnya dicari nilai dari sequence matching berdasarkan teks dokumen lalu setelahnya menampilkan tingkat similarity dengan menghitung kemamaan berdasarkan sequence matching yang didapat pada proses sebelumnya. Dan menampilkan output dari proses tersebug. Setelah itu dilanjutkan dengan mencari informasi pada dokumen, yang mana pada proses ini akan diambil informasi berdasarkan dokumen docx, pada proses ini dapat diambil informasi properties yang ada pada header docx. Lalu menampilkan output dari proses tersebuh dengan menampilkan ketiga output dari proses tersebut, maka program akan selesai.

3. HASIL DAN PEMBAHASAN

Pengujian dilakukan pada 10 file dokumen dengan ekstensi docx yang nantinya akan dilakukan sebanyak 10 kali pengujian yang nama 5 kali pengujian dengan dokumen yang berbeda, dan 5 kali pengujian dengan dokumen yang sama. Agar nantinya di ketahui berapa hasil plagiasi antara dokume yang berbeda dan dokumen yang sama persis. Analoginya jika dilakukan pengujian dengan dokumen yang sama maka hasil presentasi yang didapat adalam 100%. Dan juga pengujian dilakukan dengan varibel k, w, dan b yang berbeda beda. Jadi total pengujian yang dilakukan sebanyak 30 kali.

a. Pengujian dengan dokumen yang berbeda

Tabel 2. Hasil pengujian dokumen berbeda dengan nilai k-gram = 3, w-gram = 4 dan bilangan prima 2

Nama dokumen	Total karakter	Panjang substring	Hasil Winnowing	Hasil Ratcliff/ Obershelp	Rata-rata similarity
Dokumen 1 Dokumen 2	33 22	24	50 %	81.36 %	65.68 %
Dokumen 3 Dokumen 4	2268 6933	1155	50.24 %	25.11 %	37.67 %
Dokumen 5 Dokumen 6	3615 4417	1089	46.78 %	27.12 %	36.76 %
Dokumen 7 Dokumen 8	4161 4386	1350	41.75 %	31.59 %	36.67 %
Dokumen 9 Dokumen 10	4302 2860	402	47.05 %	11.23 %	29.41 %

Tabel 3. Hasil pengujian dokumen berbeda dengan nilai k-gram = 4, w-gram = 5 dan bilangan prima = 3

Nama dokumen	Total karakter	Panjang substring	Hasil Winnowing	Hasil Ratcliff/ Obershelp	Rata-rata similarity
Dokumen 1 Dokumen 2	33 22	24	30 %	81.36 %	55.68 %
Dokumen 3 Dokumen 4	2268 6933	1155	44.85 %	25.11 %	34.98 %
Dokumen 5 Dokumen 6	3615 4417	1089	44.58 %	27.12 %	35.9 %

Dokumen 7 Dokumen 8	4161 4386	1350	41.23 %	31.59 %	36.41 %
Dokumen 9 Dokumen 10	4302 2860	402	40.59 %	11.23 %	25.91 %

Tabel 4. Hasil pengujian dokumen berbeda dengan nilai k-gram = 5, w-gram = 6 dan bilangan prima = 4

Nama dokumen	Total karakter	Panjang substring	Hasil Winnowing	Hasil Ratcliff/ Obershelp	Rata-rata similarity
Dokumen 1 Dokumen 2	33 22	24	25 %	81.36 %	53.18 %
Dokumen 3 Dokumen 4	2268 6933	1155	21.73 %	25.11 %	22.18 %
Dokumen 5 Dokumen 6	3615 4417	1089	19.58 %	27.12 %	23.35 %
Dokumen 7 Dokumen 8	4161 4386	1350	19.80 %	31.59 %	25.69 %
Dokumen 9 Dokumen 10	4302 2860	402	15.59 %	11.23 %	13.11 %

Tabel 5. Hasil output fitur Get Document Information dari document uji

Nama Dokumen	Author	Last modified by	Created time	Last modified time
Dokumen 1	yura bee	yura bee	02/08/2022 12:09	02/08/2022 12:09
Dokumen 2	yura bee	yura bee	02/08/2022 12:09	02/08/2022 12:10
Dokumen 3	yura bee	yura bee	05/07/2022 14:46	30/07/2022 01:20
Dokumen 4	pgbwa	yura bee	05/07/2022 14:28	05/07/2022 09:36
Dokumen 5	Windows	HP	04/08/2022 15:10	04/08/2022 15:12
Dokumen 6	Acer-PC	Acer-PC	04/08/2022 15:19	04/08/2022 15:20
Dokumen 7	Pgbwa	yura bee	07/08/2022 14:31	07/08/2022 14:34
Dokumen 8		yura bee	07/08/2022 07:32	07/08/2022 07:36
Dokumen 9	Romaito	yura bee	14/07/2021 19:36	07/08/2022 14:37
Dokumen 10	Windows	Admin	07/08/2022 14:46	07/08/2022 14:46

pada tabel 5. Telah di dapatkan properties information dari masing-masing dokumen yang diuji, dengan ini, kita menyimpulkan bahwa jika, author dan date time yang sama dapat menjadi informasi tambahan bahwasanya si user telah melakukan copy paster dengan dokumen yang dibuatnya. Lalu pada dokumen 8

diketahui tidak memiliki author ini menandakan bahwa dokumen tersebut telah mengalami proses dari pihak ketiga sehingga properties informationnya menjadi reset kembali.

b. Pengujian dengan dokumen yang sama

Tabel 6. Hasil pengujian dokumen yang sama dengan nilai k-gram = 3, w-gram = 4 dan bilangan prima = 2

Nama dokumen	Total karakter	Panjang substring	Hasil Winnowing	Hasil Ratcliff/ Obershelp	Rata-rata similarity
Dokumen 1 Dokumen 1	33 33	24	100%	100%	100%
Dokumen 3 Dokumen 3	2268 2268	2268	100%	100%	100%
Dokumen 5 Dokumen 5	3615 3615	3615	100%	100%	100%
Dokumen 7 Dokumen 7	4161 4161	4161	100%	100%	100%
Dokumen 9 Dokumen 9	4302 4302	4302	100%	100%	100%

Tabel 7. Hasil pengujian dokumen yang sama dengan nilai k-gram = 4, w-gram = 5 dan bilangan prima = 3

Nama dokumen	Total karakter	Panjang substring	Hasil Winnowing	Hasil Ratcliff/ Obershelp	Rata-rata similarity
Dokumen 1 Dokumen 1	33 33	24	100%	100%	100%
Dokumen 3 Dokumen 3	2268 2268	2268	100%	100%	100%
Dokumen 5 Dokumen 5	3615 3615	3615	100%	100%	100%
Dokumen 7 Dokumen 7	4161 4161	4161	100%	100%	100%
Dokumen 9 Dokumen 9	4302 4302	4302	100%	100%	100%

Tabel 8. Hasil pengujian dokumen yang sama dengan nilai k-gram = 4, w-gram = 5 dan bilangan prima = 3

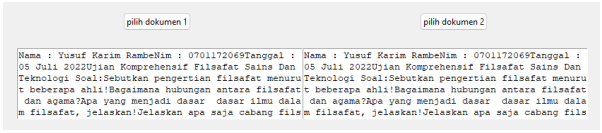
Nama dokumen	Total karakter	Panjang substring	Hasil Winnowing	Hasil Ratcliff/ Obershelp	Rata-rata similarity
Dokumen 1 Dokumen 1	33 33	24	100%	100%	100%

Dokumen 3 Dokumen 3	2268 2268	2268	100%	100%	100%
Dokumen 5 Dokumen 5	3615 3615	3615	100%	100%	100%
Dokumen 7 Dokumen 7	4161 4161	4161	100%	100%	100%
Dokumen 9 Dokumen 9	4302 4302	4302	100%	100%	100%

Pada pengujian yang di lakukan pada tabel 5,6 dan 7, didapatkan hasil dengan seluruh nilainya 100%, hal ini menandakan untuk algoritma winnowing, nilai variabel k-gram, w-gram dan bilangan prima tidak akan mempengaruhi tingkat plagiasi karna dokumen yang diuji adalah dokumen yang sama persis. Sedangkan pada algoritma Ratcliff/Obershlep. Didapatkan hasil yang serupa dikarenakan nilai dari total panjang substring sama dengan kedua dokumen yang diujikan sehingga didapatkan nilai persentase 100%.

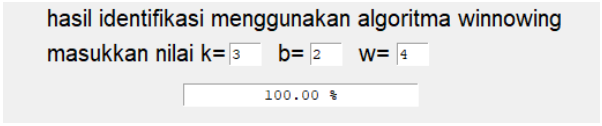
3.2 Implementasi

Berikut adalah GUI dari identifikasi tingkat kemiripan dokumen teks menggunakan fungsi hash pada algoritma winnowing dan pattern recognation pada algoritma Ratcliff/Obershelp. Gambar 5 merupakan proses dimana user akan menginput file docx yang nantinya akan di baca sistem dan pada gambar 5 akan di tampilkan teks yang nantinya akan di proses.



Gambar 5. Input dokumen

Setelah proses input dokumen dilakukan selanjutnya user akan di minta memasukkan nilai variabel k-gram, w-gram dan bilangan prima yang nantinya digunakan sebagai variabel pada algoritma yang digunakan.



Gambar 6. Input nilai k-gram, w-gram dan bilangan prima, dan hasil winnowing

Setelah nilai dari ketiga variabel dimasukkan, lalu user akan klik tombol proses dimana nantinya hasil persentasi dari algoritma winnowing dan ratfcliff akan langsung di munculkan. Untuk hasil dari algoritma Ratcliff/Obershelp dapat dilihat pada gambar berikut;

Gambar 7. Hasil Ratcliff/Obershelp

Setelah hasil dari plagiarisme di dapatkan selanjutnya secara bersamaan hasil dari get document info akan di tampilkan yang mana pada proses ini akan di tampilkan informasi properties dari masing masing dokumen yang di ujikan. Untuk contohnya dapat dilihat pada gambar beikut;

Get Document Information

properties	dokumen pertama	dokumen kedua
author	yura bee	yura bee
last_modified_by	yura bee	yura bee
last_modified_time	2022-07-30 18:20:00	2022-07-30 18:20:00
created_time	2022-07-05 07:46:00	2022-07-05 07:46:00
word_count	3716	3716
revision	4	4
last_printed	None	None

Gambar 8. Hasil Get document information

Hasil akhir dari identifikasi tingkat kemiripan pada 10 dokumen yang diujikan dapat menghasilkan persentase sampai 100% pada pengujian dengan dokumen yang sama. Dibandingkan dengan pengujian dengan dokumen yang berbeda didapatkan nilai yang bervariasi. Hasil ini cukup efektif. Dengan begitu kedua algoritma yang digunakan dapat dijadikan prospek dalam mendeteksi plagiarisme.

4. KESIMPULAN

Dari hasil yang penelitian yang telah dilakukan sebelumnya, dapat disimpukan bahwa, sistem yang dibangun menggunakan kedua algoritma winnowing dan Ratcliff/Obershelp berjalan dengan baik, dari 10 dokumen yang dilakukan pengujian diperolah tingkat kemiripan pada dokumen 1 dan 2 sebesar 65.68%, pada dokumen 3 dan 4 sebesar 37.67%. pada dokumen 5 dan 6 sebesar 36.76, pada dokumen 7 dan 8 sebesar 36.67, pada dokumen 9 dan 10 sebesar 29.41%. Dan Penerapan algoritma Winnowing dan Ratcliff/Obershelp dapat berjalan efektif pada sistem pendeteksian tingkat kemiripan dokumen. Berdasarkan hasil perhitungan manual yang cocok dengan hasil perhitugan sistem dan dari hasil dari pengujian pada dokumen yang sama memiliki tingkat kemiripan 100% . Berdasarkan hasil pengujian yang dilakukan dengan banyak karakter 22 sampai ±7000, disimpulkan bahwa banyak karakter mempengaruhi waktu eksekusi proses pendeteksian tingkat kemiripan dokumen. Penelitian selanjutnya diharapkan dapat di kembangkan agar dapat digunakan dengan berbagai ekstensi file berbasis teks, dan juga diharapkan nantinya dapat dilakukan pengujian yang dapat dilakukan pada banyak dokumen.

REFERENCE

[1]. Riki, R., Edy, E., & Maryanto, M. (2019). Plagiarism Detection Application Uses Winnowing Algorithm with Synonym Recognition for Indonesian Text Documents. Selangor Science & Technology Review (SeSTeR), 3(1),

[2]. Usman, “Kajian Plagiarisme: Studi Perbandingan Hukum Islam dan Hukum Positif di Indonesia”, Jurna Hukum dan syariah Vol. 9 No.1 2018

[3]. Sunardi, S., Yudhana, A., & Mukaromah, I. A. (2018). Implementasi Deteksi Plagiarisme Menggunakan Metode N-Gram Dan Jaccard Similarity Terhadap Algoritma Winnowing. Transmisi: Jurnal Ilmiah Teknik Elektro, 20(3)

[4]. L. J. Yudhy, Alicia S, Agustinus J. Rancang Bangun Aplikasi Deteksi Kemiripan Dokumen Teks Menggunakan Algoritma Ratcliff/Obershelp.E-journal Teknik Informatika Vol 11, No.1 (2017) ISSN: 2301-8364

[5]. Billhaqqi, T. T. I., Wicaksono, G. W., & Aditya, C. S. K. (2020). Analisis Perbandingan Algoritma Rabin-Karp Dan Winnowing Dalam Penilaian Jawaban Otomatis. In Prosiding SENTRA (Seminar Teknologi dan Rekayasa) (No. 6, pp. 269-276).

[6]. Kharisman, O., Susanto, B., & Suwarno, S. (2018). IMPLEMENTASI ALGORITMA WINNOWING UNTUK MENDETEKSI KEMIRIPAN PADA DOKUKEN TEKS. Jurnal Informatika, 9(1).

[7]. Faisal, M., Nugroho, F., M El Sulthan, M., Amini, F., Hariyadi, M. A., & Sedayu, A. (2020). Plagiarism detection using manber and winnowing algorithm. International Journal of Advanced Science and Technology, 29(6s), 2130-2136.

[8]. Nurdin, N., & Munthoha, A. (2017). Sistem Pendeteksian Kemiripan Judul Skripsi Menggunakan Algoritma Winnowing. InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan, 2(1).

[9]. Radili, A., & Sanjaya, S. (2018). Penerapan Metode Winnowing Fingerprint dan Naive Bayes untuk Pengelompokan Dokumen. Jurnal CoreIT: Jurnal Hasil Penelitian Ilmu Komputer dan Teknologi Informasi, 3(2)

[10]. Sunardi, S., Yudhana, A., & Mukaromah, I. A. (2017). Perancangan aplikasi deteksi plagiarisme karya ilmiah menggunakan algoritma winnowing.

[11]. Setiawan, A. (2017). Implementasi Algoritma Winnowing Untuk Deteksi Kemiripan Judul Skripsi Studi Kasus Stmik Budidarma. Informasi dan Teknologi Ilmiah (INTI), 4(2).

[12]. Aritonang, L. W. (2020). Rancang Bangun Aplikasi Deteksi Kemiripan Dua Gambar Menggunakan Algoritma Ratcliff/Obershelp. Journal of Computer System and Informatics (JoSYC), 1(3), 191-198.
[13]. Izzah, N., Yusliani, N., & Roodiah, D. (2022). Sistem Deteksi Kemiripan Teks Pada Berita Berbahasa Indonesia Menggunakan algoritma Ratcliff/Obershelp. Jurnal Linguistik Komputasional, 5(1), 1-6.
[14]. Hidayat, W., Utami, E., & Hartanto, A. D. (2020, November). Effect of Stemming Nazief & Adriani on the Ratcliff/Obershelp algorithm in identifying level of similarity between slang and formal words. In 2020 3rd International Conference on Information and Communications Technology (ICOIACT) (pp. 22-27). IEEE.