

Analisa Algoritma *String Matching* Dan *Winnowing* Untuk Deteksi Kemiripan Judul Tugas Akhir Perguruan Tinggi

Lilik Sugiarto, Ciske Mulyadi, Siti Rihastuti
Amikom Solo

Lilik@dosen.amikomsolo.ac.id

Abstrak

Judul Tugas akhir merupakan gambaran awal dari sebuah isi suatu dokumen, dimana judul merupakan arah awal dari sebuah isi suatu penelitian, Dimana terdapat kemiripan tugas akhir dengan isi yang diduga mirip dengan tugas akhir yang sudah pernah ada. Untuk pendeteksian awal sebuah Judul tugas akhir ada beberapa metode ataupun algoritma diantaranya *String Matching* Dan *winnowing*. Adapun *Algoritma Winnowing* merupakan salah satu metode untuk mendeteksi kesamaan (*common subsequence*). Dua teks diketahui memiliki kesamaan kata/kalimat apabila didalam dokumen tersebut dijumpai *fingerprint*, *fingerprint* inilah yang akan dijadikan dasar perbandingan antara teks, dimana nilai *fingerprint* diperoleh dari *n-gram* mengubah teks kedalam nilai angka {*hash*}. Sedangkan *string Matching* suatu metode pencocokan teks dengan langkah Memindai teks dengan bantuan sebuah *finger* yang ukurannya sama dengan panjang *pattern* kemudian Menempatkan *window* pada awal teks dan Membandingkan karakter pada *window* dengan karakter dari *pattern*. Setelah pencocokan dilakukan pergeseran ke kanan pada *window*. Prosedur ini dilakukan berulang-ulang sampai *window* berada pada akhir teks. Mekanisme ini disebut mekanisme *sliding window*. Analisa kedua metode baik *string matching* maupun *winnowing* akan dapat diketahui metode manakah yang lebih efektif dan efisien dalam deteksi dini kemiripan sebuah judul Tugas Akhir.

Kata kunci: Algoritma, *Winnowing*, *String Matching*, *Pattern*, *Fingerprint*, *hash*, *Window*

Abstract

The title of thesis is an initial description of the contents of a document, where the title is the initial direction of the contents of a research, there is a similarity between the final project and the content that is thought to be similar to an existing final project. For the early detection of a title, there are several methods or algorithms, including string matching and winnowing. The winnowing algorithm is a method for detecting similarities (common sub sequence). Two texts are known to have similar words or sentences if fingerprint are found in the document. This Fingerprint will be used as a basis for comparison between texts, where the fingerprint value obtained from n-grams converts the text into numeric values {hash}. Whereas string matching is a text matching method with the step of scanning text with the help of a finger whose size is the same as the length of the pattern then placing the window at the beginning of the text and comparing the characters in the window with the characters from the pattern. After matching is done shift right on window. This procedure is repeated until the window is at the end of the text. This mechanism is called sliding window. Analysis of both methods, both string matching and winnowing, will be able to fine out which method is more effective and efficient in early detection of the similarity of a title in the final project.

Keywords : Algorithm, *Winnowing*, *String Matching*, *Pattern*, *Fingerprint*, *hash*, *Window*

PENDAHULUAN

Salah satu ciri dalam penerapan teknologi adalah dimana teknologi dapat membantu dalam menyelesaikan permasalahan secara efektif dan efisien bagi penggunaannya. Dalam perkembangannya khususnya dalam bidang teknologi informasi telah terbukti dengan diterapkannya teknologi informasi dapat membantu memecahkan berbagai permasalahan-permasalahan yang terjadi. Adapun bidang yang telah menerapkan teknologi khususnya teknologi informasi baik itu bidang Politik, Ekonomi, sosial, Budaya serta bidang pertahanan dan keamanan suatu negara. Bahkan bisa dikatakan salah satu ciri sebuah negara maju tercermin dalam seberapa besar tingkat penerapan teknologi khususnya Teknologi informasi. Hal inilah salah satu bukti bahwa penerapan teknologi salah satu cara dalam menyelesaikan permasalahan secara efektif dan efisien.

Dalam dunia pendidikan khususnya perguruan tinggi penerapan teknologi informasi buakanlah suatu hal yang asing dewasa ini, bahkan peran teknologi informasi suatu perguruan tinggi teknologi informasi dijadikan suatu sarana yang vital dalam membangun civitas akademika yang berdaya saing antara perguruan tinggi yang satu dengan perguruan tinggi lain baik secara nasional maupun skala internasional.

STMIK AMIKOM SURAKARTA adalah salah satu perguruan tinggi di wilayah surakarta yang baru berkembang saat ini. STMIK AMIKOM SURAKARTA yang berlokasi di jalan Veteran Notosuman, Kel Singopuran, kec. Kartasura Kab. Sukoharjo Prop.Jawa Tengah di beberapa tahun belakangan ini sedang giat membangun sebuah sistem informasi akademik. Harapannya Pembangunan sistem informasi ini dapat menjadi salah satu daya saing dengan perguruan tinggi yang lain, sesuai visi dan misi perguruan tinggi yaitu menjadi perguruan tinggi unggulan dalam bidang ilmu pengetahuan dan teknologi informasi tahun 2030. Adapun pengembangan layanan yang berbasis teknologi informasi yang sudah diterapkan pada STMIK AMIKOM SURAKARTA belum menasar dalam hal layanan pengajuan judul Tugas akhir secara online. Penulis tertarik dengan koleksi tugas akhir di perpustakaan STMIK AMIKOM SURAKARTA diduga terdapat kemiripan judul, hal inilah

menurut penulis akan berdampak buruk baik perkembangan ilmu pengetahuan itu sendiri maupun secara institusi yang disebabkan tugas akhir yang hanya membahas permasalahan yang sama sehingga kurangnya khasanah keragaman judul Tugas Akhir.

Sebelum penerapan teknologi informasi pengajuan judul secara online ini diterapkan maka perlu adanya analisa lebih mendalam mengenai metode ataupun algoritma apa saja yang nantinya dapat digunakan dalam rancang bangun sebuah sistem informasi pengajuan judul tugas akhir berbasis web pada STMIK AMIKOM SURAKARTA. Kajian ini perlu dilakukan guna mengetahui metode apa yang tepat untuk mengatasi permasalahan ini dan salah satu dari.

Dalam kasus permasalahan ini penulis mencoba untuk membuat penelitian awal dengan menganalisa beberapa algoritma tentang perbandingan kata ataupun kalimat dengan algoritma *STRING MATCHING* dan *WINNOWER* untuk Deteksi awal kemiripan sebuah Tugas Akhir dengan harapan kedepan semakin banyak dan beraneka raga bahasan sebuah tugas akhir tentang teknologi informasi. Dengan ini penulis membuat penelitian yang berjudul *ANALISA ALGORITMA STRING MATCHING DAN WINNOWER UNTUK DETEKSI KEMIRIPAN JUDUL TUGAS AKHIR PERGURUAN TINGGI*.

Rumusan Masalah

Dalam penelitian ini dapat dirumuskan bahwa bagaimana Analisa Algoritma *STRING MATCHING* Dan *Winnower* Untuk Deteksi Kemiripan Judul Tugas Akhir Perguruan Tinggi yang kedepannya dapat dijadikan sebuah acuan awal dalam rancang bangun sebuah sistem informasi tugas akhir secara online.

1.3. Tujuan Penelitian

Tujuan penelitian ini adalah menerapkan ilmu bidang teknologi informasi dan mengetahui beberapa algoritma dalam pembobotan guna mengetahui tingkat kemiripan sebuah kata ataupun kalimat yang nantinya difokuskan untuk penilaian kemiripan judul tugas akhir

METODE

Pada penelitian yang diusulkan mengambil studi kasus di STMIK AMIKOM SURAKARTA yang berlokasi di jalan Veteran Notosuman, Kel Singopuran, kec. Kartasura Kab. Sukoharjo

Prop.Jawa Tengah, adapun Metode yang digunakan dalam penelitian ini adalah *action research*, yaitu bertujuan untuk mengembangkan pendekatan baru dalam memecahkan masalah dengan penerapan langsung di dunia nyata ((Hasibuan, 2007) Hazibuan, 2007).

3.1 Metode Pengumpulan Data

Berdasarkan jenis datanyadata yang dikumpulkan dalam penelitian ini adalah sebagai berikut:

3.1.1 Data Primer

Data primer adalah data yang diambil langsung dari objek penelitian atau merupakan data yang berasal dari sumber asli atau pertama. Teknik pengumpulan data primer dilakukan melalui teknik observasi dan wawancara.

Untuk teknik observasi pada penelitian ini digunakan untuk mengetahui prosedur langkah kerja sistem pengajuan judul Tugas Akhir yang sudah berjalan. Serta memperoleh data tentang kendala-kendala dalam melakukan langkah kerja sistem pengajuan judul Tugas Akhir secara konvensional tersebut.

Teknik wawancara pada penelitian ini peneliti melakukan wawancara kepada pejabat pejabat yang berwenang yang sesuai dengan keahlian bidang Akademik pada STMIK AMIKOM SURAKARTA. Hal ini dilakukan untuk mendukung objek penelitian supaya hasil penelitian sesuai dengan yang diharapkan

3.1.2 Data Sekunder

Data sekunder adalah data yang tidak didapatkan secara langsung dari objek penelitian, melainkan data yang berasal dari sumber yang telah dikumpulkan dari pihak lain. Teknik pengumpulan data sekunder dilakukan dengan cara studi dokumentasi dan studi literatur.

3.2 Metode Analisis Data

3.2.1 Algoritma String Matching

String matching merupakan salah satu Algoritma pencarian sebuah kata ataupun kalimat yang biasa dikenal dengan istilah pencocokan *string*, cara kerja algoritma ini dalam menyelesaikan permasalahan pencocokan *string* dengan melakukan pencarian kemunculan *string* pendek yang dikenal dengan istilah *pattern*(0..n-1) dan sedangkan *string* yang lebih panjang yang disebut teks (0..m-1). Dalam teknik untuk

memecahkan permasalahan pencocokan *string* ini terdapat beberapa algoritma diantaranya, *Algoritma string matching Brute Force*, *Algoritma Rabin Karp*, *Algoritma Knout Morris Pratt* dan *Algoritma Boyer moore*(Waruwu & Mandala, 2016).

Dari beberapa algoritma *string matching* diatas pada dasarnya setiap algoritma memiliki kelebihan dan kekurangan masing-masing. Pada penelitian yang dilakukan(Waruwu & Mandala, 2016)dapat disimpulkan dari kedua algoritma algoritma *KMP (Knout Morris Pratt)* dan algoritma Boyer Moore tersebut algoritma boyer moore diketahui membutuhkan waktu lebih cepat dalam akses pencocokan *string*". Dari hasil tersebut diatas penulis kecenderungan tertarik menggunakan algoritma *KMP (Knout Morris pratt)*dalam penyelesaian masalah pencocokan *string* dimana dengan algoritma tersebut memiliki beberapa kelebihan.

string matching dirumuskan dengan :

$$x = x [0 \dots m - 1] \dots\dots\dots(2.1)$$

$$y = y [0 \dots n - 1] \dots\dots\dots(2.2)$$

Dimana:

- x = *Pattern*
- m = panjang *Pattern*
- y = teks
- n = panjang teks

Menurut (Effendi et al., 2013) Prinsip kerja algoritma *string matching* adalah sebagai berikut :

1. Pemindaian Teks
 Dalam algoritma *string matching* tahapan pertaman yaitun dengan Memindai teks dengan bantuan sebuah *window* yangukurannyasamadengan panjang *Pattern*. Dicontohkan algoritma Knouth morris pratt dalam pencocokan
 Pattern = INFO
 Teks = SISTEM INFORMASI
2. Menempatkan *window* pada awal teks.
 Langkah berikutnya yaitu pencocokan digambarkan pada tabel 1.1 dibawah ini. Dimana data *Pattern* dan teks ditempatkan kedalam *window* yang nantinya akan dilakukan pergeseran atau dengan istilah *sliding window*.

Tabel 3.1 langkah ke 1

Pattern	I	N	F	O											
Teks	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I
indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Membandingkan karakter pada *window* dengan karakter dari *Pattern*. Setelah pencocokan dilakukan pergeseran ke kanan pada *window*.

Tabel 3.2 langkah ke 2

Pattern		I	N	F	O										
Teks	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I
indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Tabel 3.3 langkah ke 3

Pattern			I	N	F	O									
Teks	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I
indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Tabel 3.4 langkah ke 4

Pattern				I	N	F	O								
Teks	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I
indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Tabel 3.5 langkah ke 5

Pattern					I	N	F	O							
Teks	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I
indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Tabel 3.6 langkah ke 6

Pattern						I	N	F	O						
Teks	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I
indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Tabel 3.7 langkah ke 7

Pattern							I	N	F	O					
Teks	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I
indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

3.2.2 Algoritma Winnowing

Algoritma *winnowing* adalah salah satu algoritma pencocokan *string* dimana dalam pendekteksianya harus memenuhi kebutuhan mendasar (Astutik et al., 2014) yaitu

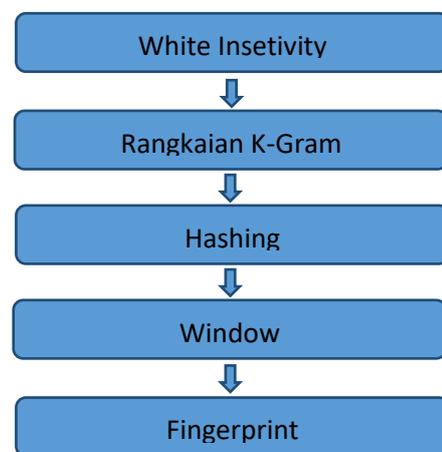
1. *Whitespace insetivity* yaitu pencarian kalimat yang tidak terpengaruh oleh spasi, jenis huruf(

kapital atau normal), tanda baca dan sebagainya.

2. *Noise supresion* yaitu menghindari penemuankecocokan dengan panjang kata yang terlalu kecil atau kurang relevan seperti “the” dan bukan kata yang umum digunakan.
3. *Position independece* yaitu penemuan kesamaanharus tidak bergantung pada posisi kata-kata sehingga kata dengan urutan posisi berbeda masih dapat dikenali jika terjadi kesamaan.

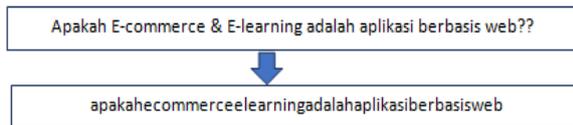
Proses pengidentifikasian kemiripan sebuah dokumen dapat dilakukan dengan menggunakan algoritma *Winnowing*. Algoritma ini menggunakan metode document *fingerprinting* untuk medeteksi keakuratan salinan antar dokumen atau hanya sebagian teks.

Secara umum algoritma *winnowing* memiliki beberapa konsep dalam melakukan pendeteksiian sebuah dokumen, berikut konsep kerja algoritma *winnowing*:



Gambar 3.1 konsep algoritma *Winnowing*.

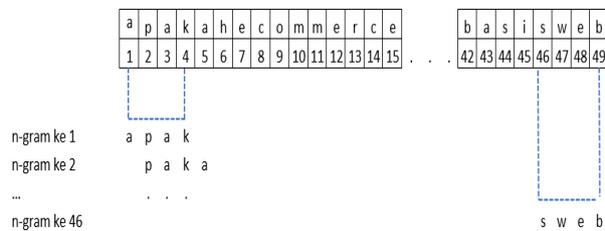
- 3.2.2.1 Penghapusan karakter-karakter yang tidak relevan (*whitespace insensitivity*). Pada tahap Penghapusan karakter yang tidak relevan ini algoritma *winnowing* melakukan proses penghapusan tanda baca, spasi dan simbol atau karakter seperti !, @, #, \$, %, ^, &, *, (,), _ , -, +, <, >, ?berikut sebagai contoh: E-commerce & E-learning adalah bentuk aplikasi berbasis web!!



Gambar 3.2 contoh pengapusan karakter

3.2.2.2 Pembentukan rangkaian n-gram.

Proses pembentukan rangkaian *n-gram* pada algoritma *winning* dilakukan dengan cara membentuk rangkaian karakter sepanjang *n* yang diambil dari hasil penghapusan karakter yang tidak diperlukan, rangkaian *n-gram* pertama kali dimulai dari karakter ke 1 sampai dengan karakter *n*. Nilai *n* yang baik tidak terlalu kecil dan tidak teralubesar (Syahputra, 2017). Berikut contoh bentuk nilai *n-gram* dengan panjang *n* = 4 :



Contoh lain Hasil n-gram:

s/ste istem stemi temin
eminf minfo infor rform
forma omas rmasi masib
asibe siber iberb berba
erbas rbasi basis asiw

3.2.2.3 Penghitungan nilai hash setiap n-gram.

Algoritma *winning* menggunakan *rolling hash* untuk menghitung nilai *hash* masing-masing rangkaian gram.

Berikut rumus fungsi *hash* dan *rolling hash*

$$H_{(C_1...C_n)} = C_1 * b^{(n-1)} + C_2 * b^{(n-2)} + \dots + C_{(n-1)} * b^{(n)} + C_n \dots \dots \dots (3.1)$$

$$H_{(C_2...C_{(n+1)})} = (H_{(C_1...C_n)} - C_1 * b^{(n-1)}) * b + C_{(n+1)} \dots \dots \dots (3.2)$$

Dimana:

$$H_{(C_1...C_n)} = \text{Nilaihash}$$

C_1 = Nilai ASCII karakter ke -1 pada *string*

n = Panjang *string*

b = nilai basis *hash*

Berikut contoh Hasil *rolling hash*

83506	89554	88480	80035
84612	82591	84959	81129
86932	87990	83755	78213
88056	81084	77313	80407
86615	76880	78315	88589

3.2.2.4 Membagi ke dalam *window*.

Algoritma *winning* tidak menggunakan nilai *hash* yang dibentuk pada tahap sebelumnya akan dibagi ke dalam *window* berukuran *w*. berikut contoh pembentukan *window* dari hasil perhitungan nilai *hash* pada tahap sebelumnya dengan ukuran lebar *window* (*w*) = 4.

w=4

83506	89554	88480	80035
89554	88480	80035	84612
88480	80035	84612	82591
80035	84612	82591	84959
84612	82591	84959	81129
82591	84959	81129	86932
84959	81129	86932	87990
81129	86932	87990	83755
86932	87990	83755	78213
87990	83755	78213	88056
83755	78213	88056	81084

3.2.2.5 Pemilihan beberapa nilai *hash* menjadi *document fingerpringting*.

Setelah terbentuk *window* seluruh nilai *hash*, tahap berikutnya yaitu dengan menentukan nilai *fingerprint* teks. Nilai *fingerprint* ditentukan dengan memilih *hash* terkecil dari setiap *window*. Pemilihan nilai *fingerprint* dari hasil pembentukan *window* pada tahap sebelumnya.

83506	89554	88480	80035
89554	88480	80035	84612
88480	80035	84612	82591
80035	84612	82591	84959
84612	82591	84959	81129
82591	84959	81129	86932
84959	81129	86932	87990

Maka *fingerprint* yang terbentuk yaitu

80035 81129 78213 77313 76880 78315

3.2.2.6 Persamaan jaccard Coeficient.

Nilai *Fingerprint* terbentuk dari algoritma *winning* digunakan untuk mengukur prosentase tingkat kemiripan teks pada persamaan(Syahputra, 2017).

Adapun rumus persamaan jaccard Coeficient.

$$similarity = \frac{jumlah\ fingerprint\ sama}{Total\ seluruh\ fingerprint} \times 100\%$$

.....(3.

3)

atau

$$similiar(d_i, d_j) = \frac{|w(d_i) \cap w(d_j)|}{|w(d_i) \cup w(d_j)|} \times 100\%.... (3.4)$$

Dengan nilai d_i nilai- nilai *fingerprint* pada teks, d_j nilai-nilai *fingerprint* pada teks $w(d_i) \cap w(d_j)$ jumlah nilai *fingerprint* yang sama antara teks ke -i dan teks ke-j dan $w(d_i) \cup w(d_j)$ adalah total nilai *fingerprint* teks ke-i dan teks ke-j.

3.3 Alur Penelitian

Desain penelitian ini mengikuti siklus *action research*, Langkah-langkah dalam *action research* menurut Baskerville dalam lee(2007), yaitu *Diagnosing, action planning, action taking, evaluating, dan reflection*.

a. Diagnosing

Tahap ini merupakan proses identifikasi masalah – masalah yang menjadi dasar penelitian tentang deteksi dini kemiripan judul tugas akhir. Pada tahap ini juga dilakukan proses pengumpulan data, data yang dikumpulkan adalah data primer dan data sekunder. Data primer diperoleh melalui observasi secara langsung yaitu dengan melakukan observasi ke STMIK AMIKOM fokus pada unit akademik yang menangani pengajuan judul tugas akhir sedangkan data sekunder diperoleh melalui studi literatur dengan menggali informasi jurnal penelitian terbaru tentang deteksi kemiripan judul Tugas akhir.

b. Action Planning

Tahap ini merupakan proses penyusunan rencana tindakan yang tepat guna menyelesaikan permasalahan pada penelitian ini. Tindakan Tindakan yang direncanakan meliputi penerapan kaidah *String Matching*

dan *Winnowing*. Tahap ini juga membahas desain sistem yang akan diterapkan. Desain yang digunakan uuntuk alur sistem menggunakan *Flow chart*.

c. Action Taking

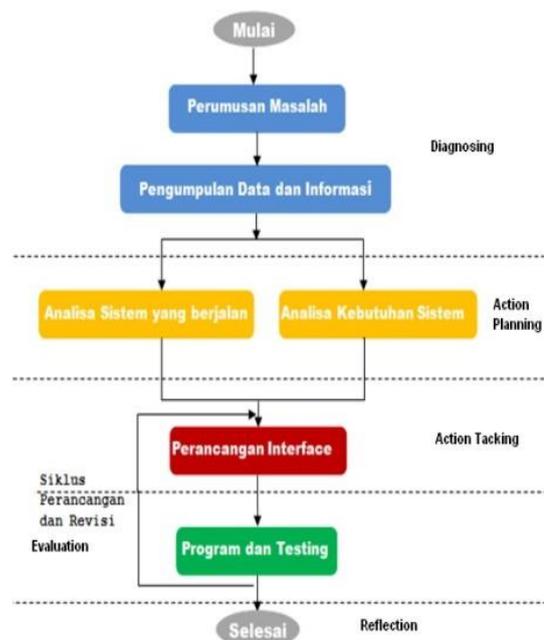
Tahap ini merupakan tahapan implementasi desain pada tahap Action planning ke dalam program atau aplikasi guna melakukan perhitungan atau pembobotan..

d. Evaluation

Tahap ini merupakan tahapan dimana sistem diuji secara menyeluruh. Tujuan utama dari pengujian sistem adalah untuk memastikan bahwa integrasi antar modul aplikasi telah memenuhi spesifikasi kebutuhan dan berjalan sesuai dengan skenario pengujian, kemudian menguji dengan masukkan tertentu yang diharapkan menghasilkan keluaran yang sesuai.

e. Reflection

Tahap ini merupakan bagian akhir siklus yang telah dilaluidengan melaksanakan review tiap tahapdalam penelitian ini penelitian dapat berakhir. AdapunLangkah-langkah dalam *action research* seperti terlihat dallam gambar dibawah ini.



Gambar 3.3 alur penelitian

$$\begin{aligned}
 &+ascii(t) * 5^{(1)} + ascii(e) \\
 &= 115*625+105*125+115*25+116*5+101 \\
 &= 88556 \\
 H_{(istem)} &= ascii(s) * 5^{(4)} + ascii(i) * 5^{(3)} \\
 &\quad + ascii(s) * 5^{(2)} \\
 &\quad + ascii(t) * 5^{(1)} + ascii(e) \\
 &= 105*625+115*125+116*25+101*5+109 \\
 &= 83514
 \end{aligned}$$

Dengan menggunakan hasil pembentukan rangkaian-rangkaian n-gram sebelumnya, dengan menggunakan basis (b)=5 dan panjang n-gram (n)=5, perhitungan hash yaitu:

88556	83514	89550	88485	80027	84621
83748	78216	88069	81068	77312	80425

4.1.2.4 Pembentukan *window* dari nilai *hash*

Algoritma *winnowing* tidak menggunakan semua nilai *hash* dari setiap rangkaian gram yang dibentuk pada tahap sebelumnya akan dibagi ke dalam *window* berukuran w. *Window* pertama berisi nilai *hash* ke -w+1 dan seterusnya sampai terbentuk *window* dari seluruh nilai *hash*. Pembentukan *window* dari hasil perhitungan nilai *hash* pada tahap sebelumnya dengan ukuran lebar *window* (w)=4 yaitu:

<i>window</i> ke 1	88556	83514	89550	88485
<i>window</i> ke 2	83514	89550	88485	80027
<i>window</i> ke 3	89550	88485	80027	84621
<i>Window</i> ke 4	88485	80027	84621	82594
<i>window</i> ke 5	80027	84621	82594	84954
<i>window</i> ke 6	84621	82594	84954	81117
<i>window</i> ke 7	82594	84954	81117	86950
<i>window</i> ke 8	84954	81117	86950	87980
<i>window</i> ke 9	81117	86950	87980	83748
<i>window</i> ke 10	86950	87980	83748	78216
<i>window</i> ke 11	87980	83748	78216	88069
<i>window</i> ke 12	83748	78216	88069	81068
<i>window</i> ke 13	78216	88069	81068	77312
<i>window</i> ke 14	88069	81068	77312	80425
<i>window</i> ke 15	81068	77312	80425	86605
<i>window</i> ke 16	77312	80425	86605	76890
<i>window</i> ke 17	80425	86605	76890	78319
<i>window</i> ke 18	86605	76890	78319	88571

window ke 19 76890 78319 88571 83578

4.1.2.5 Pemilihan *Fingerprint* dari setiap *window*
Setelah terbentuk *window* dari seluruh nilai *hash*, maka tahap selanjutnya adalah menentukan nilai *Fingerprint* teks. Nilai *fingerprint* ditentukan dengan memilih nilai *hash* terkecil dari setiap *window*. Pemilihan nilai *fingerprint* dari hasil pembentukan *window* pada tahap sebelumnya yaitu:

[88556	83514	89550	88485]
[83514	89550	88485	80027]
[89550	88485	80027	84621]
[88485	80027	84621	82594]
[80027	84621	82594	84954]
[84621	82594	84954	81117]
[82594	84954	81117	86950]
[84954	81117	86950	87980]
[81117	86950	87980	83748]
[86950	87980	83748	78216]
[87980	83748	78216	88069]
[83748	78216	88069	81068]
[78216	88069	81068	77312]
[88069	81068	77312	80425]
[81068	77312	80425	86605]
[77312	80425	86605	76890]
[80425	86605	76890	78319]
[86605	76890	78319	88571]
[76890	78319	88571	83578]

4.1.2.6 Persamaan *Jaccard Coeficient*

Nilai *Fingerprint* terbentuk dari algoritma *winnowing* digunakan untuk mengukur prosentase tingkat kemiripan teks pada persamaan (Syahputra, 2017).

Adapun rumus persamaan jaccard Coeficient.

$$\begin{aligned}
 &similarity \\
 &= \frac{jumlah\ fingerprint\ sama}{Total\ seluruh\ fingerprint} \times 100\%
 \end{aligned}$$

Dari persamaan ini akan diambil jumlah nilai *fingerprint* yang sama dan dibagi dengan total nilai *fingerprint* maka bisa dihitung tingkat kemiripannya dengan derajat prosentase(%).

Dimana nilai *fingerprint* diperoleh dari dua judul Tugas Akhir yang berbeda yang dihitung tingkat

kemiripannya dengan menggunakan algoritma *winnowing*.

Judul1 = **sisteminformasiberbasisweb**

Judul2 =**sistempenjualanberbasisweb**

Berikut hasil penghitungan kemiripan dengan rumus jaccard Coeficient

fingerprint1	83514	80027	81117	78216	77312	76890
fingerprint2	83514	80165	80207	77323	77312	76890

$$\begin{aligned} \text{similarity} &= \frac{6}{12} \times 100\% \\ &= 50\% \end{aligned}$$

Berdasarkan perhitungan tingkat kemiripan kedua judul tugas akhir yang diterjemahkan ke dalam nilai *fingerprint* menggunakan algoritma *winnowing* maka kemiripan antara judul 1 dengan judul2 memiliki tingkat kemiripan sebesar 50%.

KESIMPULAN

Analisa algoritma antara *string matching* dan *winnowing* untuk deteksi kemiripan judul Tugas Akhir telah dapat dilakukan dengan mengetahui kinerja kedua algoritma tersebut. Hasil penelitian sebagai berikut:

1. Hasil pengujian kemiripan judul dengan menggunakan *string matching* dengan algoritma Knout Morris Pratt berhasil menentukan kemiripan *string* atau teks
2. Untuk efektifitas algoritma *string matching* tingkat kemiripannya belum terlihat jelas dalam derajat prosennya(%).
3. Algoritma *String matching* lebih efisien dalam proses penghitungan jika dibandingkan dengan Algoritma *Winnowing*.
4. Algoritma *winnowing* untuk proses penghitungannya lebih kompleks jika dibandingkan dengan algoritma *string matching*.
5. Kelebihan algoritma *winnowing* yaitu bisa menghitung secara otomatis tingkat kemiripannya dalam bentuk prosentase.

DAFTAR PUSTAKA

1. Astutik, S., Cahyani, A. D., & Sophan, M. K. (2014). Sistem Penilaian Esai Otomatis Pada E-Learning Dengan Algoritma *Winnowing*. *Jurnal Informatika*, 12(2), 47–52. <https://doi.org/10.9744/informatika.12.2.47-52>
2. Effendi, D., Hartono, T., & Kurnaedi, A. (2013). Penerapan String Matching Menggunakan Algoritma Boyer-Moore Pada Translator Bahasa Pascal Ke C. *Majalah Ilmiah UNIKOM*, 11(2), 262–275.
3. Hasibuan, Z. A. (2007). Metodologi Penelitian Pada Bidang Ilmu Komputer Dan Teknologi Informasi. *Konsep, Teknik, Dan Aplikasi, Universitas Indonesia*, 194.
4. Sinaga, J. I., Mesran, M., & Buulo, E. (2016). Aplikasi Mobile Pencarian Kata Pada Arti Ayat Al-Qur ' an Berbasis Android Menggunakan ... *Jurnal INFOTEK*, II(Juni 2016), 68–72.
5. Sunardi, S., Yudhana, A., & Mukaromah, I. A. (2018). Implementasi Deteksi Plagiarisme Menggunakan Metode N-Gram Dan Jaccard Similarity Terhadap Algoritma *Winnowing*. *Transmisi*, 20(3), 105. <https://doi.org/10.14710/transmisi.20.3.105-110>
6. Syahputra, A. R. (2017). Implementasi Algoritma *Winnowing* Untuk Deteksi Kemiripan Judul Skripsi Studi Kasus: STMIK Budi Darma. *Pelita Informatika Budi Darma*, 12(1), 1–9.
7. Waruwu, F. T., & Mandala, R. (2016). Perbandingan Algoritma Knuth Morris Pratt dan Boyer Moore Dalam Pencocokan String Pada Aplikasi

Kamus Bahasa Nias. *Jurnal Ilmiah*
INFOTEK, 1(1), 36–43.

.