

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Hasil penelitian terdahulu seperti pada tabel 2.1 memberikan gambaran tentang penelitian yang akan dilakukan sebagai pedoman maupun acuan penulisan dalam melakukan penelitian guna menambah wawasan pengetahuan dan menambah bahan bacaan objek penelitian yang sedang dibuat.

Tabel 2. 1 Tabel penelitian terdahulu

No	Judul	Penulis (Tahun)	Isi	Perbedaan
1	Deteksi Plagiarisme Tingkat Kemiripan Judul Skripsi dengan Algoritma WInnowing	Nur Alamsyah (2017)[13]	Pada penelitian ini membahas pengajuan judul skripsi mahasiswa yang akan diajukan agar tidak sama dengan yang telah ada.	Perbedaan penelitian ini dengan penelitian yang akan dibuat adalah di penggunaan algoritma. Algoritma pada penelitian ini menggunakan WInnowing sedangkan algoritma di penelitian yang akan dibuat menggunakan Horspool. Untuk

			objek di kedua penelitian ini sama yaitu kemiripan judul skripsi
2	Penerapan Algoritma Horspool untuk Apikasi Pencarian Abstrak Skripsi Berbasis Web	Elisa Malat, Eka Arriyanti dan hanifah Eka Wati (2015)[10]	Penelitian ini membahas tentang pencarian abstrak skripsi untuk kajian empiris dari penelitian yang dikerjakan dengan menggunakan algoritma Horspool pada abstrak skripsi. Untuk metode yang digunakan sama, yaitu menggunakan algoritma Horspool.
3	Perancangan Aplikasi untuk Menghitung Presentase Kemiripan Proposal dan Isi Skripsi dengan Algoritma Rabin-Karp	Ledisma Juliana Purba dan Lambot Sitorus (2018)[14]	Pada penelitian ini menjelaskan perancangan aplikasi pendeteksi plagiasi guna menghindari kesamaan judul, naskah pada proposal dan isi dari skripsi masing-masing metode

		masing mahasiswa.	menggunakan algoritma Rabin-Karp. Untuk penelitian yang akan dilakukan menggunakan objek judul skripsi dan untuk metode menggunakan algoritma Horspool.
4	Penerapan Algoritma Horspool pada Aplikasi Katalog Perpustakaan	Fince Tinus Waruwu dan Rivalri Kristianto Hondro (2019)[12]	<p>Pada penelitian ini menjelaskan banyaknya buku dalam perpustakaan tidak mungkin jika dicari manual. Dari masalah tersebut dibangun sebuah aplikasi pencarian jenis buku yang dibutuhkan.</p> <p>Pada penelitian ini, perbedaan dengan penelitian yang akan dibuat adalah di objek. Penelitian ini menggunakan jenis buku sedangkan penelitian yang akan dibuat menggunakan judul skripsi. Untuk metode yang digunakan sama yaitu algoritma Horspool</p>

5	Pembuatan Sistem Pendeteksi Kemiripan Pengajuan Topik Skripsi Menggunakan Metode Rabin-Karp	Ahmad Husein Siti Nur Salam (2018)[2]	Penelitian ini menjelaskan bahwa dalam menulis tugas kuliah, tugas akhir berpotensi melakukan penjiplakan. Termasuk juga dalam pengajuan topik skripsi.	Perbedaan terletak pada objek dan penggunaan metode. Di penelitian ini menggunakan metode Rabin-Karp dan objek skripsi, sedangkan pada penelitian yang akan dibuat menggunakan metode Horspool dan objek judul skripsi.
---	---	---------------------------------------	---	---

Pada penelitian yang akan dilakukan untuk mendeteksi kemiripan judul skripsi menggunakan Algoritma Horspool. Kemiripan judul bisa dilakukan dengan pergeseran dan pencocokan antar karakter. Pencocokan inilah yang nantinya bisa untuk mendeteksi bahwa judul skripsi tersebut mirip atau tidak dengan yang sudah ada. Nantinya output yang dihasilkan berupa sistem pendeteksi kemiripan judul skripsi dengan jumlah presentase kemiripan.

2.2 Landasan Teori

2.2.1. Plagiarisme

Plagiarisme merupakan tindakan yang dilakukan baik sengaja ataupun tidak sengaja untuk memperoleh nilai-nilai dari karya ilmiah dengan mengutip sebagian atau keseluruhan tanpa

memberikan sumber dengan tujuan menjadikan karya tersebut hak milik sendiri[15]. Literasi informasi termasuk salah satunya mengenai penghindaran kegiatan plagiarisme tidak hanya dilakukan oleh mahasiswa dan dosen saja namun dimulai sejak dini yaitu pada siswa dan guru[16].

2.2.2. String Matching

String matching merupakan suatu pendekatan yang berfungsi untuk menemukan kecocokan antar dua string berbeda, yaitu string pattern dan string text. String pattern dan string text terdiri dari kumpulan karakter yang disebut dengan kata alfabet yang disimbolkan dengan Σ (sigma) dan memiliki ukuran σ (tao) [17]. Pencocokan string juga dapat digunakan untuk mencari bitdalam sejumlah besar file binary, dalam algoritma string matching, teks diasumsikan berada didalam memory, sehingga bila mencari string di dalam sebuah teks, maka semua isi teks perlu dibaca terlebih dahulu kemudian disimpan didalam memorystring matching fokus pada pencarian satu, atau lebih umum, semua kehadiran sebuah kata (lebih umum disebut pattern) dalam sebuah teks. Semua kehadiran yang dibahas menegelurkan semua kehadiran pola dalam teks, pola dinotasikan sebagai $x = x[0..m-1]$; m adalah panjangnya[18].

2.2.3. Algoritma Horspool

Algoritma Horspool adalah sebuah metode pencarian teks (string matching) yang mencocokkan pola string dari kanan ke kiri dan bergeser sesuai tabel nilai pergeseran yang telah ditentukan[10]. Algoritma horspool hampir sama kerjanya dengan algoritma boyer moore dimana melakukan kesamaan antara pattern yang dikasih indeks dengan teks. Algoritma Horspool tidak efisien untuk kata yang pendek. Pada algoritma horspool diberikan nilai pergeseran karakter dari yang sebelah kanan jendela. Pada pengamatan dari yang pertama diberikan nilai pergeseran yang

akan diukur pada setiap karakter. Selanjutnya dilakukan persamaan pattern mulai ke dari kanan baru kekiri sampai terdapat hasil kesimpulan sama atau tidak sama[12].

Menurut [17] terdapat dua tahap pada pencocokan string menggunakan algoritma Horspool, yaitu :

1. Tahap praproses, pada tahap ini dilakukan observasi pattern terhadap teks untuk membangun sebuah tabel bad-match yang berisi nilai shift ketika ketidakcocokan antara pattern dan teks terjadi. Secara sistematis, langkah-langkah yang dilakukan algoritma Horspool pada tahap praproses adalah:

- a. Algoritma Horspool melakukan pencocokan karakter terkanan pada pattern.
- b. Setiap karakter pada pattern ditambah ke dalam tabel badmatch dan dihitung nilai shiftnya.
- c. Karakter yang berada pada ujung pattern tidak dihitung dan tidak dijadikan karakter terkanan dari karakter yang sama dengannya.
- d. Apabila terdapat dua karakter yang sama dan salah satunya bukan karakter terkanan, maka karakter dengan indeks terbesar yang dihitung nilai shift-nya.
- e. Algoritma Horspool menyimpan panjang dari pattern sebagai panjang nilai shift secara default apabila karakter pada teks tidak ditemukan dalam pattern.
- f. Nilai (value) shift yang akan digunakan dapat dicari dengan perhitungan panjang dari pattern dikurang indeks terakhir karakter dikurang 1, untuk masing-masing karakter, $value = m - I - 1 \dots \dots \dots (2.1)$

2. Tahap pencarian Secara sistematis, langkah-langkah yang dilakukan algoritma *Horspool* pada tahap praproses adalah:

- a. Dilakukan perbandingan karakter paling kanan *pattern* terhadap *window*.

- b. Tabel *bad-match* digunakan untuk melewati karakter ketika ketidakcocokan terjadi.
 - c. Ketika ada ketidakcocokan, maka karakter paling kanan pada *window* berfungsi sebagai landasan untuk menentukan jarak *shift* yang akan dilakukan.
3. Proses perhitungan Horspool

Proses perhitungan Algoritma Horspool dimulai dengan menentukan setiap karakter dengan nilai Index. Nilai index diawali mulai angka 0 dan berakhir sampai dengan jumlah karakter pada kalimat tersebut.

Pattern : MUSA				
	M	U	S	A
	0	1	2	3

Gambar 2. 1 Menentukan indeks karakter[19]

Pada Gambar 2.1 terdapat angka 0, 1, 2, dan 3. Angka tersebut adalah nilai indeks pada setiap karakter dalam kalimat MUSA. Setelah menentukan nilai indeks, selanjutnya adalah menghitung nilai Value. Nilai value didapat dengan menggunakan rumus (2.1)

$value = 4 - 0 - 1 = 3$ $value = 4 - 1 - 1 = 2$ $value = 4 - 2 - 1 = 1$ * : karakter yang tidak dikenali

Gambar 2. 2 menghitung value[19]

Setelah nilai value terhitung seperti pada gambar 2.2, langkah selanjutnya adalah membuat tabel Bad Match yang berisi nilai indeks dan nilai value

Karakter	Index	Value
M	0	3
U	1	2
S	2	1
*	-	4

Gambar 2. 3 tabel Bad Match praproses[19]

Gambar 2.3 menunjukkan tabel Bad Match yang berisi nilai indeks dan nilai value. Proses selanjutnya adalah tahap pencarian secara sistematis. Pada awal tahap ini adalah pembuatan tabel Bad Match, dapat dilihat pada gambar 2.4.

P	M	U	S	A
i	0	1	2	3
V	3	2	1	4

Gambar 2. 4 tabel BadMatch di pencarian sistematis[19]

Setiap teks dan pattern masing-masing diberi nilai m dan i , dimana m sebagai panjang pattern dan i sebagai indeks. Gambar 2.4 menunjukkan nilai pergeseran bad-match dengan menghitung nilai v seperti yang telah dilakukan pada gambar 2.2. Tahap selanjutnya adalah tahap awal pencarian, dilakukan pencocokan karakter paling kanan pattern terhadap window. Apabila terjadi ketidakcocokan, akan dilakukan pergeseran ke kanan untuk melewati karakter yang tidak cocok dimana nilai pergeserannya terdapat pada tabel bad-match. Karakter paling kanan teks pada window berfungsi sebagai landasan untuk menentukan jarak geser yang akan dilakukan. Hal ini terlihat pada Gambar 2.5 berikut:

<i>m</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P	M	U	S	A												
<i>i</i>	0	1	2	3												

Gambar 2. 5 pencarian sistematis 1[19]

Pada Gambar 2.5 ketidakcocokan karakter “Spasi” dan “A”. karakter “Spasi” tidak terdapat pada Tabel Bad-Match sehingga diganti oleh tanda (*). Tanda (*) Bernilai sebesar 4 sehingga dilakukan pergeseran sebanyak 4 kali. Hal ini terlihat pada Gambar 2.6.

<i>m</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P					M	U	S	A								
<i>i</i>					0	1	2	3								

Gambar 2. 6 pencarian sistematis 2[19]

Pada Gambar 2.6, terdapat ketidakcocokan kembali antara karakter “I” dan “A”. Karakter “I” tidak terdapat pada Tabel Bad-Match sehingga diganti oleh tanda (*). Tanda (*) bernilai sebesar 4 sehingga dilakukan pergeseran sebanyak 4 kali. Terlihat pada Gambar 2.7

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P										M	U	S	A			
i										0	1	2	3			

Gambar 2. 7 pencarian sistematis 3[19]

Pada Gambar 2.7 menunjukkan ketidakcocokan pattern dan teks terjadi pada karakter “S” dan “A”. Karakter “S” terdapat pada Tabel bad-match yang bernilai sebesar 2 sehingga dilakukan pergeseran sebanyak 2 kali. Hal ini terlihat terlihat pada Gambar 2.8.

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P										M	U	S	A			
i										0	1	2	3			

Gambar 2. 8 pencarian sistenatis 4[19]

Pada Gambar 2.8, window telah berada pada teks dan semua pattern cocok dengan teks. Seluruh pencocokan karakter menggunakan algoritma horspool telah selesai dan berhenti pada pencarian sistematis keempat.

2.2.4. Jaccard Similarity

Jaccard Similarity atau Jaccard Coefficient merupakan algoritma yang fungsinya untuk membandingkan dua sample yaitu dokumen yang satu dengan yang lainnya berdasarkan kata yang dimilikinya. Jaccard similarity biasanya digunakan untuk membandingkan dokumen dan menghitung nilai kemiripan (similarity) dari dua buah objek atau dokumen[20]. Rumus Jaccard Similarity ditunjukkan pada (2.2)

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \dots\dots\dots(2.2)$$

$A \cap B$ = Jumlah kata yang sama (\cup = Union)

$A \cup B$ = Jumlah seluruh kata (\cap = Intersection)

Pada perhitungan Jaccard, nilai $|A \cap B|$ merupakan jumlah fingerprint yang sama antara dokumen A dengan dokumen B. Untuk dapat mengetahui nilai dokumen A sama dengan dokumen B dilakukan penyimpanan setiap nilai pada dokumen A kemudian dibandingkan dengan setiap nilai pada dokumen B, apabila sesuai maka nilai irisan ditambahkan dan disimpan. Bila semua nilai sudah dibandingkan maka proses berhenti dan nilai $|A \cap B|$ sudah diketahui. Pada proses $|A \cup B|$ bisa dihitung dengan mencari nilai jumlah fingerprint pada dokumen A kemudian ditambah dengan jumlah fingerprint pada dokumen B[21].

Misalkan terdapat dua buah dokumen yang dilakukan pembobotan dan pengindeksan. Isi dari kedua dokumen yang digunakan tersebut dapat dirincikan, dapat dilihat pada gambar 2.9 sebagai berikut:

Dokumen 1 = sarjana teknik informatika unissula
 Dokumen 2 = sarjana teknik sipil unissula

Gambar 2. 9 Contoh dua buah dokumen [21]

Kemudian memisahkan dua dokumen di atas menjadi array; (1)sarjana, (2)teknik, (3)informatika, (4)sipil , (5)unissula. Berarti memiliki dua set yang berbeda yaitu Dokumen 1 dan Dokumen 2.

Dokumen 1 = A

Dokumen 2 = B

$$A = \{ 1,2,3,5\} \text{ dan } B = \{1,2,4,5\} \dots\dots\dots(2.3)$$

Kemudian mencari Union dari kedua dokumen tersebut. Union adalah jumlah kata secara keseluruhan dari dua dokumen yang sedang dihitung. Dari array diatas bisa dilihat bahwa jumlah kata secara keseluruhan adalah 5 kata. Union dari Dokumen 1 dan 2 adalah sebagai berikut :

$$A \cup B = \{1,2,3,4,5\} \dots\dots\dots(2.4)$$

Setelah berhasil mendapatkan hasil Union pada persamaan (2.4), selanjutnya adalah mencari Intersection diantara dua dokumen tersebut. Intersection adalah jumlah kata yang sama dari dua dokumen yang sedang dihitung. Jika dilihat dari Dokumen A dan Dokumen B, ada beberapa kata yang sama dari kedua dokumen tersebut, antara lain : [1]sarjana, [2]teknik , [5]unissula. Intersection dari Dokumen A dan B adalah :

$$A \cap B = \{1,2,5\} \dots\dots\dots(2.5)$$

Langkah selanjutnya adalah menghitung kemiripan dari kedua dokumen tersebut dengan rumus (2.2)

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{3}{5} = 0,6$$

Berdasarkan nilai persamaan yang diperoleh maka dapat ditetapkan bahwa nilai kemiripan dari dokumen 1 dan dokumen 2 adalah 0,6. Hasil tersebut dikalikan dengan 100 karena untuk mencari presentase kemiripan, jadi presentase dari kedua dokumen tersebut sebesar 60%.

2.2.5. MySQL

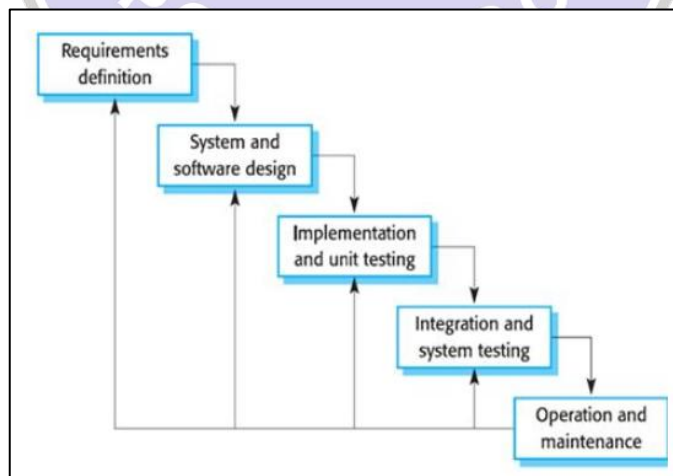
MySQL adalah salah satu jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang databse sebagai sumber dan pengelolaan datanya[22]. MySQL bisa dikatakan perangkat lunak sistem manajemen basis data SQL atau DBMS yang multithread, multi-user dengan sekitar

6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License [23]. MySQL merupakan database yang awalnya hanya berjalan pada sistem Unix dan Linux. Seiring berjalannya waktu dan banyaknya peminat yang menggunakan database ini, MySQL merilis versi yang dapat diinstal pada hampir semua platform, termasuk Windows. Lisensi dari MySQL adalah freeware[24]..

2.2.6. PHP MyAdmin

PHP merupakan kependekan dari Hypertext Preprocessor. PHP tergolong sebagai perangkat lunak open source yang diatur dalam aturan general purpose licences (GPL). Bahasa pemrograman PHP sangat cocok dikembangkan dalam lingkungan web, karena PHP bisa diletakkan pada script HTML atau sebaliknya. PHP dikhususkan untuk pengembangan web dinamis[25]. Karena PHP bersifat open source, siapa pun dapat menggunakannya tanpa dipungut biaya. Pemrograman PHP memungkinkan program dijalankan pada sistem operasi apa pun dengan browser web[26].

2.2.7. Metode Waterfall



Gambar 2. 10 Metode Waterfall

Metode waterfall seperti pada gambar 2.10 merupakan model pengembangan sistem informasi yang sistematis dan sekuensial[27]. Metode Waterfall memiliki tahapan-tahapan sebagai berikut :

1. Requirements analysis and definition

Layanan sistem, kendala, dan tujuan ditetapkan oleh konsultasi dengan pengguna dan menjelaskan secara rinci spesifikasi sistem

2. Sistem and software design

Perancangan tampilan tatap muka secara keseluruhan baik perangkat keras maupun perangkat lunak. Perancangan perangkat melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat.

3. Implementation and unit testing

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

4. Integration and sistem testing

Unit-unit pada perangkat digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan atau tidak. Setelah pengujian, perangkat lunak dapat dikirimkan ke pengguna..

5. Operation and maintenance

Tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. Maintenance melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.

2.2.8. White Box Testing

Metode ini untuk pengujian aplikasi atau sistem. Pengujian ini banyak dilakukan karena metode ini sangat baik untuk pengujian. Pengujian White Box telah berkembang sebuah analisis sistem yang dapat menggambarkan secara tepat kode sumber pemrograman, prosedur sistem, dan jalur. Metode pengujian White box diimplementasikan guna menentukan kapasitas sistem yang diciptakan bisa bekerja secara efisien dan sistematis. Pengujian white box merupakan cara pengujian yang menerapkan tatanan control yang sesuai prosedur. Pengujian ini dapat digunakan untuk menganalisa aplikasi setelah implementasi sistem[28]

