

## TEXT MINING UNTUK MENDETEKSI PLAGIASI DOKUMEN DENGAN PENERAPAN STEMMING NAZIEF-ADRIANI DAN ALGORITMA SMITH-WATERMAN

Alvika Meitaningsih<sup>(1)</sup>, Agus Sasmito Aribowo<sup>(2)</sup>, Nur Heri Cahyana<sup>(3)</sup>

Jurusan Teknik Informatika Fakultas Teknik Industri UPN "Veteran" Yogyakarta

Jl. Babarsari 2 Tambakbayan Yogyakarta

e-mail : meitahadi78@gmail.com<sup>(1)</sup>, sasmito.skom@upnyk.ac.id<sup>(2)</sup>, dsnurheri@gmail.com<sup>(3)</sup>

### Abstract

*Plagiarism is the act of copying someone else's work and acknowledging it as a result of his personal work. There are currently many algorithms that discuss how to detect text document plagiarism such as Cosine, Smith Waterman. The results of previous studies stated that the Smith Waterman algorithm has low accuracy, so that in this study the development of the Smith Waterman Algorithm was carried out. The Smith Waterman algorithm is commonly used in the field of bioinformatics to determine the similarity of DNA, but in this study the Smith Waterman algorithm can be implemented to detect documents. The process of detecting document similarities is first carried out by the preprocessing process to eliminate affixes to facilitate the process of document detection using stemming. Stemming used in this study was Stemming Nazief & Adriani and to measure the level of accuracy in the document detection process, calculations were made using the Smith Waterman algorithm to obtain the percentage similarity between documents. From the trials carried out the addition of preprocessing, stemming affects the time of the testing process because in this preprocessing process the words that are affixed will be returned to the base words and matched with dictionary data in the database.*

**Keywords : Document Plagiarism, Smith Waterman, Stemming**

### Abstrak

Plagiarisme adalah tindakan menjiplak karya orang lain dan mengakui sebagai hasil karya pribadinya. Saat ini sudah banyak algoritma yang membahas cara mendeteksi plagiarisme dokumen teks seperti *Cosine, Smith Waterman*. Hasil penelitian sebelumnya menyatakan bahwa algoritma *Smith Waterman* memiliki keakurasian yang rendah, sehingga pada penelitian ini dilakukan pengembangan dari Algoritma *Smith Waterman*. Algoritma *Smith Waterman* biasa digunakan didalam bidang bioinformatika untuk menentukan kesamaan DNA, akan tetapi dalam penelitian ini Algoritma *Smith Waterman* dapat diimplementasikan untuk mendeteksi dokumen. Proses pendeteksian kemiripan dokumen pertama-tama dilakukan proses *preprocessing* untuk menghilangkan imbuhan guna memudahkan proses pendeteksian dokumen yaitu dengan menggunakan *stemming*. *Stemming* yang digunakan dalam penelitian ini adalah *Stemming* Nazief & Adriani dan untuk mengukur tingkat keakurasian pada proses pendeteksian dokumen dilakukan perhitungan menggunakan algoritma *Smith Waterman* untuk mendapatkan hasil persentase kemiripan antar dokumen. Dari uji coba yang dilakukan penambahan *preprocessing* yaitu *stemming* mempengaruhi waktu proses pengujian karena pada proses *preprocessing* ini kata yang berimbuhan akan dikembalikan ke kata dasar dan dicocokkan dengan data kamus yang ada didalam *database*.

**Kata kunci: Plagiarisme Dokumen, Smith Waterman, Stemming**

## 1. PENDAHULUAN

Dalam perkuliahan mahasiswa Teknik Informatika diwajibkan mengambil mata kuliah sesuai dengan tingkatan semester yang ditempuh dan mahasiswa biasanya akan mendapatkan tugas sesuai dengan mata kuliah yang ditempuh berupa tugas kelompok maupun individu. Dalam proses pembuatan tugas masing-masing mahasiswa banyak yang masih sering meng-kopi hasil tugas dari hasil tugas orang lain, oleh karena itu dengan memanfaatkan perkembangan teknologi informasi ini suatu sistem pengujian yang efektif dan efisien dalam melakukan pengujian dokumen secara lebih cepat, tepat dan menghindari kesamaan dokumen tersebut. Namun,

pendeteksian kemiripan teks menjadi pekerjaan yang sulit dilakukan oleh manusia karena banyak dan besarnya teks untuk dibandingkan serta strukturnya yang tidak konsisten dan kompleks. Pendeteksian kemiripan teks dapat dilakukan untuk berbagai tujuan, salah satunya adalah untuk mencegah plagiarisme. Proses memeriksa tugas yang dilakukan secara manual memang hasilnya akan lebih akurat, akan tetapi membutuhkan waktu yang lebih lama diperlukan membaca satu persatu dokumen tugas untuk menilai dan mengecek dokumen tersebut.

Salah satu tanda adanya tindakan plagiarisme adalah terdapat kesamaan antar dokumen dengan dokumen yang menjadi patokan atau sumber yang dijadikan sebagai acuan. Dalam tindakan plagiarisme terdapat kesamaan isi dokumen hanya saja dapat dilakukan dengan cara mengganti kata atau merubah kata dan membolak-balikan susunan kalimat atau mengurangi kalimat pada dokumen yang dijadikan sebagai acuan. Untuk mengatasi teks yang berukuran besar dan banyak diperlukan sebuah aplikasi untuk mengotomasi pendeteksian plagiasi untuk mempercepat proses pemeriksaan dokumen. Fokus dalam penelitian ini adalah pengimplementasian proses *Text Mining* yang digunakan untuk melakukan klasifikasi untuk menemukan pola dari sekumpulan tekstual yang berjumlah besar dan untuk mendeteksi struktur teks yang tidak konsisten dan kompleks.

Karena dokumen yang akan diuji menggunakan bahasa Indonesia penelitian ini mengimplementasikan *Stemming* Nazief-Adriani untuk mengubah kata-kata yang menggunakan bahasa Indonesia menjadi kata-kata dasarnya (*stemming*) yang kemudian akan digunakan dalam proses selanjutnya yaitu tahap preprocessing dan Algoritma Smith-Waterman yang berfungsi untuk menghitung kemiripan teks berdasarkan urutan kata. Dalam penelitian ini menggunakan Algoritma untuk mendapatkan hasil akurasi yang maksimal pada pendeteksian plagiasi dokumen. Selain mendapatkan hasil yang lebih akurat penggunaan algoritma ini bertujuan untuk mengklasifikasi kesamaan dokumen berdasarkan isi dan susunan dari dokumen tersebut, sehingga dapat ditemukan keterkaitan atau relasi antara isi dokumen satu dengan dokumen yang dijadikan sebagai pembandingan. Klasifikasi dilakukan dengan cara mengelompokkan data ke dalam kelas-kelas yang sudah diberikan berdasarkan kesamaan sifat dan pola yang terdapat pada data tersebut kemudian data dibandingkan berdasarkan kesamaan isi dengan membandingkan dengan kelompok-kelompok dokumen yang sudah ada sehingga akan ditemukan kedekatan antar dokumen.

## 2. TINJAUAN PUSTAKA

Penelitian Radiant Victor Imbar, dkk yang berjudul "*Implementasi Cosine Similarity dan Algoritma Smith-Waterman Untuk Mendeteksi Kemiripan Teks*" membahas mengenai pendeteksian kemiripan dokumen dengan metode Algoritma Cosine Similarity dan Smith-Waterman untuk menangani teks berukuran besar yang banyak, maka dengan mengotomatiskan proses pendeteksian kemiripan teks dengan menggunakan konsep text mining. Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi yang mengimplementasikan cosine similarity yang berguna untuk mengukur kesamaan teks berdasarkan kemunculan kata-kata dalam teks tersebut dan algoritma Smith-Waterman yang berfungsi untuk menghitung kemiripan teks berdasarkan urutan kata.

Penelitian Shandy Ardiansyah Abdurrasyid, dkk yang berjudul "*Implementasi dan Optimasi Algoritma Nazief-Adriani Untuk Stemming Dokumen Bahasa Indonesia*" membahas mengenai pendeteksian kemiripan dokumen yaitu dengan menggunakan Algoritma Nazief-Adriani yang digunakan untuk proses pemotongan imbuhan-imbuhan suatu kata menjadi kata dasarnya dengan proses *stemming*. Dan setelah dilakukan pengujian hasil menunjukkan bahwa Algoritma ini menunjukkan hasil yang lebih akurat.

Penelitian Ilham Candra Budiman (2009). Tugas Akhir. Universitas Indonesia dengan judul "*Penyejajaran Lokal Barisan DNA dengan Menggunakan Metode Smith-Waterman*" Penelitian ini membahas tentang pencarian pasangan segmen dari dua barisan DNA yang telah disejajarkan, dua barisan belum tentu memiliki similaritas secara keseluruhan tetapi bisa saja hanya mengandung pasangan segmen dari kedua barisan yang memiliki kemiripan yang tinggi. Metode

---

yang digunakan untuk memaksimalkan hasil pencarian pasangan segmen tertinggi adalah Metode Smith-Waterman.

Penelitian Farid Bangkit Djafar. Jurnal. UNG, dengan judul “Penerapan Algoritma Smith-Waterman dalam Sistem Pendeteksian Kesamaan Dokumen” Penelitian ini membahas tentang pendeteksian kesamaan dokumen dengan membandingkan dua dokumen dalam sebuah matriks. Dari matriks ini nantinya akan dilakukan traceback untuk mencari letak kesamaan dan apakah terdapat penghapusan dan penyisipan dari dokumen aslinya. Hasil yang diperoleh pengguna adalah seberapa besar presentasi dari kesamaan dokumen dan menunjukkan letak kesamaannya. Algoritma ini berhasil diterapkan untuk system pendeteksian dokumen.

Penelitian Helmy Darmawan (2009). Tugas Akhir. Universitas Telkom, dengan judul “Analisis dan Implementasi Algoritma Smith-Waterman pada Proses Identifikasi Kesamaan Dokumen” Penelitian ini membahas tentang pendeteksian untuk mengetahui kemiripan yang signifikan (local alignment) dari suatu sequence gen yang berasal dari tubuh makhluk hidup dengan cara yang disebut dengan sequence alignment (penyejajaran sekuens). Dengan mengimplementasikan Algoritma Smith-Waterman tersebut sebagai alat bantu dalam menentukan kesamaan dokumen teks, setiap kalimat pada dokumen teks akan dianalogikan sebagai sebuah sekuens gen yang disusun oleh rantai-rantai basa higrorn yang berupa kata. Selain itu, untuk dapat menyesuaikan masukan dan meningkatkan

Penelitian Fariz Ashar Himawan (2013). Tugas Akhir. Institut Pertanian Bogor dengan judul “Penyejajaran Lokal Sekuens DNA dengan Menggunakan Algoritma Smith-Waterman” Penelitian ini membahas tentang penerapan Algoritma Smith-Waterman dan membandingkan dengan hasil yang sudah pernah dilakukan oleh Utomo (2013) dan dilakukan validasi similaritas dan gaps yang dihasilkan menggunakan EMBOSS-water sebagai software penyejajaran local yang berbasis Algoritma Smith-Waterman, sehingga hasil yang didapatkan dari penyejajaran local lebih optimal.

Penelitian Arif Bambang Prasetya (2017). Jurnal. Institut Teknologi Bandung dengan judul “Penerapan Algoritma Dinamis pada Penyejajaran Sekuens dengan Algoritma Smith-Waterman” Penelitian ini membahas tentang metode dari bioinformatika yaitu menyusun sekuens sehingga dapat disamakan dengan sekuens lainnya, hal ini digunakan untuk mendapatkan hubungan fungsional, structural, atau evolusioner diantar 2 sekuens tersebut. Tidak hanya dalam sekuens biologi, metode ini dapat digunakan untuk menghitung biaya jarak edit diantara string dalam natural language atau dalam ata finansial. Meskipun sekuens yang sangat pendek atau mirip dapat dijajarkan dengan manual atau dengan tangan. Masalah yang sangat dibutuhkan diperlukan penyejajaran yang panjang, dan memiliki sekuens yang berjumlah sangat banyak sehingga tidak dapat dijajarkan oleh kemampuan manusia. Sebagai gantinya pengetahuan manusia dapat digunakan untuk membuat algoritma yang dapat menghasilkan penyejajaran sekuens yang optimal. Salah satunya adalah Algoritma Smith-Waterman yang berdasarkan Algoritma program dinamis yang digunakan untuk menentukan hasil yang lebih optimal meskipun hasilnya sedikit lambat.

### 3. METODOLOGI PENELITIAN

#### 3.1. Metode pengembangan sistem

Perancangan sistem menggunakan metode GRAPPLE. Metode ini merupakan pemodelan proses pengembangan perangkat lunak (*software*) dengan menitik beratkan pada aksi-aksi yang dilakukan disejumlah tahap, dimana setiap tahap akan menghasilkan (*output*) dengan bentuk yang berorientasi objek (Schmuller, 1999.)

Terdapat empat tahap dalam metodologi GRAPPLE yang digunakan dalam tugas akhir ini yaitu:

1. *Requirement Gathering*
2. *Analysis*
3. *Design*
4. *Development*
5. *Deployment*

#### Metode Smith Waterman

Algoritma Smith Waterman bukan merupakan penyejajaran global melainkan merupakan penyejajaran local. Algoritma Smith Waterman dapat mencari pasangan segmen dari dua barisan yang memiliki tingkat kemiripan paling besar dibandingkan dengan pasangan segmen yang lainnya.

Berikut ini contoh dua barisan:

$X = x_1x_2x_3 \dots x_n - 1$  dengan panjang karakter n, dan barisan

$Y = y_1y_2y_3 \dots y_m - 1$  dengan panjang karakter m

Pada contoh kedua barisan tersebut, langkah selanjutnya adalah mencari pasangan dari kedua segmen tersebut yang memiliki tingkat kemiripan paling besar.

1. Langkah pertama untuk mencari pasangan dari kedua segmen tersebut adalah membuat tabel berukuran n+1 dan m+1.

**Tabel 1. Skor Awal**

	0	X1	X2	...	Xn-1	Xn
0	H0,0	H0,1	H0,2	...	H0,n-1	H0,n
Y1	H1,0	H1,1	H1,2	...	H1,n-1	H1,n
Y2	H2,0	H2,1	H2,2	...	H2,n-2	H2,n
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
Ym-1	Hm-1,0	Hm-1,1	Hm-1,2	...	Hm-1,n-1	Hm-1,n
Ym	Hm,0	Hm,1	Hm,2	...	Hm,n-1	Hm,n

Langkah yang selanjutnya adalah menentukan pembobotan pada Algoritma Smith Waterman, pada penyejajaran barisan terdapat skor penyejajaran yang menentukan tingkat kemiripan dari dua data barisan. Skor penyejajaran dalam metode ini merupakan total skor dari pemasangan antar karakter pada kedua segmen. Skor pemasangan yang digunakan yaitu:

$X = x_1x_2x_3 \dots x_n - 1$  dengan panjang karakter n, dan barisan

$Y = y_1y_2y_3 \dots y_m - 1$  dengan panjang karakter m

Maka bobot pada pemasangan karakter  $x_i$  pada barisan X dengan  $y_j$  pada barisan Y, dimana  $1 \leq i \leq n$  dan  $1 \leq j \leq m$

$$S(x_i, y_j) = 1.0 \text{ jika } x_i = y_j$$

$$= -0.3 \text{ jika } x_i \neq y_j$$

$W_k$  = bobot pemasangan karakter pada barisan dengan spasi yang memiliki panjang k =  $1.0 + 0.3 * k$  dimana k adalah panjang spasi.

Contoh :

Pada penyejajaran berikut ini:

Adenine Guanine Thymine Guanine Thymine  
 \_            \_            Adenine            \_            Guanine Adenine

Bobot dari pemasangan karakter T pada barisan pertama dengan A pada barisan kedua Adenine Guanine **Thymine** Guanine Guanine Thymine

\_            \_            **Adenine**            \_            Guanine Adenine

Skornya adalah  $-0.3$  ( $x_i \neq y_j$ )

Bobot pemasangan karakter G pada barisan pertama dengan G pada barisan kedua

Adenine Guanine Thymine Guanine **Guanine** Thymine  
 \_            \_            Adenine            \_            **Guanine** Adenine

Skornya adalah 1.0  $(x1 = y1)$

Contoh pemasangan barisan dengan spasi bobot dari penyejajaran

**Adenine Guanine** Thymine Guanine Guanine Thymine  
 Adenine Guanine Adenine  
 Skornya adalah  $1.0 + 0.3 * 2 = 1 + 0.6 = 1.6$

- Langkah yang kedua adalah inialisasi pada Algoritma Smith Waterman dengan cara membuat tabel skor seperti Tabel 2.

**Tabel 2. Skor Inialisasi**

	0	X1	X2	...	Xn-1	Xn
0	H0,0	H0,1	H0,2	...	H0,n-1	H0,n
Y1	H1,0	H1,1	H1,2	...	H1,n-1	H1,n
Y2	H2,0	H2,1	H2,2	...	H2,n-2	H2,n
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
Ym-1	Hm-1,0	Hm-1,1	Hm-1,2	...	Hm-1,n-1	Hm-1,n
Ym	Hm,0	Hm,1	Hm,2	...	Hm,n-1	Hm,n

Inialisasi pada Algoritma Smith Waterman

$H_{0,0}$  = skor maksimum sebelumnya adanya pemasangan karakter.  
 = 0

$H_{k,0}$  = pemasangan k karakter awal pada barisan X dengan spasi.  
 = 0

Dalam pemasangan tersebut menandakan tidak adanya kesamaan pada pemasangan segmen k karakter awal pada barisan X dengan spasi sepanjang k pada barisan Y maka skor pemasangannya bernilai 0 sehingga pemasangan tersebut tidak mempengaruhi nilai dari skor penyejajaran segmen yang lain.

Contoh :

Pada penyejajaran barisan:

X = Adenine Adenine Adenine Guanine Guanine Thymine

Y =        \_        \_        \_        Guanine Guanine Thymine

Skor penyejajaran dari penyejajaran tersebut merupakan total skor dari skor sebelum adanya pemasangan karakter ditambah dengan skor pemasangan tiga karakter Adenine pada barisan X dengan tiga spasi pada barisan yang Y ditambah skor pemasangan Guanine dari barisan X dengan karakter Guanine dari barisan Y ditambah skor pemasangan karakter Thymine dari barisan X dengan karakter Thymine dari barisan Y sehingga skor penyejajaran menjadi:

$$0 - (1 + 0.3 * 3) + 1 + 1 + 1 = 0 - 1,9 + 3 = 1,1$$

Pasangan segemen yang memiliki karakter dengan kemiripan paling tinggi adalah:

Guanine Guanine Thymine

Guanine Guanine Thymine

Skor penyejajarannya adalah:

$$1 + 1 + 1 = 3$$

Skor penyejajaran dari pasangan segmen

Guanine Guanine Thymine

Guanine Guanine Thymine merupakan bagian dari penyejajaran dari :

Adenine Adenine Aadenine Guanine Guanine Thymine  
 Guanine Guanine Thymine

Skor akan tetap bernilai 3 maka pasangan dari 3 karakter awal pada barisan tersebut yang berpasangan dengan spasi sepanjang 3 harus diberi nilai 0.

$H_{0,j}$  = Pemasangan 1 karakter awal pada barisan Y dengan spasi.  
 $= 0$

Similar dengan inialisasi dari  $H_{k,0}$  karena pemasangan tersebut menandakan tidak adanya kesamaan pada pemasangan segmen k karakter awal pada barisan Y dengan spasi sepanjang l pada barisan X maka skor bernilai 0 sehingga pemasangan tersebut tidak mempengaruhi nilai dari skor penyejajaran segmen yang lain.

$H_{i,j}$  = Skor maksimum dari penyejajaran dua segmen yang berakhir di  $x_i$  dan  $y_j$

Formula untuk  $H_{i,j}$

$$H_{i,j} = \max \{H_{i-1,j-1} + s(x_i, y_j), \max_{k \geq 1} (H_{i-k,j} - W_k), \max_{l \geq 1} (H_{i,j-l} - W_l), 0\}$$

$$1 \leq i \leq n \text{ dan } 1 \leq j \leq m$$

Formula tersebut terjadi dikarenakan pada penyejajaran yang berakhir pada  $x_i$  dari barisan X dan  $y_j$  dari barisan Y terdapat empat kemungkinan yaitu:

- o Pemasangan terakhir dari penyejajaran tersebut adalah karakter  $x_i$  dengan karakter  $y_j$  sehingga skor penyejajaran yang berakhir pada  $x_{i-1}$  dan  $y_{j-1}$  ditambahkan dengan skor pemasangan  $x_i$  dan  $y_j$ :

$$H_{i,j} = H_{i-1,j-1} + s(x_i, y_j)$$

- o Pemasangan terakhir dari penyejajaran tersebut adalah pemasangan k karakter pada barisan X dengan spasi sepanjang k pada barisan Y sehingga skor penyejajaran adalah skor penyejajaran yang berakhir pada  $x_{i-k}$  dan  $y_j$  dikurangi skor pemasangan k karakter dengan spasi

$$H_{i,j} = (H_{i-k,j} - W_k)$$

- o Pemasangan terakhir dari penyejajaran tersebut adalah pemasangan l karakter pada barisan Y dengan spasi sepanjang l pada barisan X sehingga skor penyejajaran adalah skor penyejajaran yang berakhir pada  $x_i$  dan  $y_{j-1}$  dikurangi skor pemasangan l karakter dengan spasi.

$$H_{i,j} = (H_{i,j-l} - W_l)$$

- o Jika ketiga pilihan diatas tidak lebih besar dari nilai 0 maka penyejajaran segmen yang berakhir pada  $x_i$  dan  $y_j$  dinyatakan tidak memiliki kemiripan, agar pasangan segmen tersebut tidak mempengaruhi skor penyejajaran segmen lain maka skor penyejajaran pasangan segmen tersebut diberi nilai 0.

$$H_{i,j} = 0$$

### 3. Penelusuran Kembali

Setelah pengisian tabel skor selesai hingga pemasangan karakter yang terakhir maka akan dilakukan penelusuran kembali  $H_{i,j}$  merupakan skor penyejajaran maksimum dari pemasangan dua segmen dari dua barisan sehingga semakin besar nilai  $H_{i,j}$  maka kemiripan dari dua segmen tersebut semakin besar. Berdasarkan hal tersebut maka pasangan segmen yang memiliki kemiripan paling besar akan memiliki nilai  $H_{i,j}$  terbesar dari semua nilai  $H$  yang ada, misalnya terdapat pasangan segmen dengan kemiripan terbesar memiliki nilai  $H_{i,j}$  maka segmen tersebut berakhir pada  $x_i$  dan  $y_j$  sehingga penelusuran kembali dimulai dari  $H_{i,j}$ , kemudian setelah tahu bahwa pemasangan segmen dengan kemiripan paling tinggi berakhir pada  $x_i$  dan  $y_j$  selanjutnya dicari letak pasangan segmen tersebut dimulai. Pada penelusuran kembali dimulai dari nilai  $H_{i,j}$  yang paling besar. Penelusuran kembali dilakukan sampai ditemukan  $H_{i,j}$  yang bernilai 0 yang menandakan segmen yang berakhir pada nilai  $H_{i,j}$  tersebut memiliki kemiripan sehingga pasangan segmen tersebut tidak termasuk dalam pemasangan segmen yang memiliki kemiripan terbesar karena akan membuat kemiripan dari pasangan tersebut berkurang.

**Perhitungan Hasil Similarity**

Barisan pertama : Saya Kuliah di Teknik Informatika

Barisan kedua : Saya Kuliah Teknik Informatika

Yang pertama harus dilakukan adalah membuat tabel skor dari penyejajaran tersebut:

**Tabel 3. Matriks H Awal**

	0	Saya	Kuliah	Teknik	Informatika
0	$H_{0,0}$	$H_{0,1}$	$H_{0,2}$	$H_{0,3}$	$H_{0,4}$
Saya	$H_{1,0}$	$H_{1,1}$	$H_{1,2}$	$H_{1,3}$	$H_{1,4}$
Kuliah	$H_{2,0}$	$H_{2,1}$	$H_{2,2}$	$H_{2,3}$	$H_{2,4}$
di	$H_{3,0}$	$H_{3,1}$	$H_{3,2}$	$H_{3,3}$	$H_{3,4}$
Teknik	$H_{4,0}$	$H_{4,1}$	$H_{4,2}$	$H_{4,3}$	$H_{4,4}$
Informatika	$H_{5,0}$	$H_{5,1}$	$H_{5,2}$	$H_{5,3}$	$H_{5,4}$

Kemudian langkah selanjutnya mencari nilai  $H_{i,j}$  :

$$H_{i,j} = 0,0$$

$$H_{k,0} = 0,0$$

$$H_{0,l} = 0,0$$

$$H_{1,1} = \max \{H_{1-1,1-1} + s(x_1, y_1), \max (H_{1-1,1}-W_1), \max (H_{1,1-1}-W_1), 00\} = 1.0$$

$$H_{2,1} = \max \{H_{1,0} + s(x_2, y_1), \max (H_{2-1,1}-W_1, H_{2-2,1}-W_2), \max (H_{1,1-1}-W_1), 00\} = 0.0$$

$$H_{3,1} = \max \{H_{2,0} + s(x_3, y_1), \max (H_{3-1,1}-W_1, H_{3-2,1}-W_2), \max (H_{3-3,1}-W_3), \max (H_{1,1-1}-W_1), 00\} = 0.0$$

$$H_{4,1} = \max \{H_{3,0} + s(x_4, y_1), \max_{k \geq 1} (H_{4-k,1}-W_k), \max_{l \geq 1} (H_{4,1-l}-W_l), 00\} = 0.0$$

$$H_{5,1} = \max \{H_{4,0} + s(x_5, y_1), \max_{k \geq 1} (H_{5-k,1}-W_k), \max_{l \geq 1} (H_{5,1-l}-W_l), 00\} = 0.0$$

$$H_{1,2} = \max \{H_{0,1} + s(x_1, y_2), \max (H_{1-1,2}-W_1), \max (H_{1,2-1}-W_1, H_{1,2-2}-W_2), 00\} = 0.0$$

$$H_{2,2} = \max \{H_{1,1} + s(x_2, y_2), \max (H_{2-1,2}-W_1, H_{2-2,2}-W_2), \max (H_{2,2-1}-W_1, H_{2,2-2}-W_2), 00\} = 2.0$$

$$H_{3,2} = \max \{H_{2,1} + s(x_3, y_2), \max_{k \geq 1} (H_{3-k,2}-W_k), \max_{l \geq 1} (H_{3,2-l}-W_l), 00\} = 0.7$$

$$H_{4,2} = \max \{H_{3,1} + s(x_4, y_2), \max_{k \geq 1} (H_{4-k,2}-W_k), \max_{l \geq 1} (H_{4,2-l}-W_l), 00\} = 1.0$$

$$H_{5,2} = \max \{H_{4,1} + s(x_5, y_2), \max_{k \geq 1} (H_{5-k,2}-W_k), \max_{l \geq 1} (H_{5,2-l}-W_l), 00\} = 0.1$$

$$H_{1,3} = \max \{H_{0,2} + s(x_1, y_3), \max_{k \geq 1} (H_{1-k,3}-W_k), \max_{l \geq 1} (H_{1,3-l}-W_l), 00\} = 0.1$$

$$H_{2,3} = \max \{H_{1,2} + s(x_2, y_3), \max_{k \geq 1} (H_{2-k,3}-W_k), \max_{l \geq 1} (H_{2,3-l}-W_l), 00\} = 1.0$$

$$H_{3,3} = \max \{H_{2,2} + s(x_3, y_3), \max_{k \geq 1} (H_{3-k,3}-W_k), \max_{l \geq 1} (H_{3,3-l}-W_l), 00\} = 1.7$$

$$H_{4,3} = \max \{H_{3,2} + s(x_4, y_3), \max_{k \geq 1} (H_{4-k,3}-W_k), \max_{l \geq 1} (H_{4,3-l}-W_l), 00\} = 1.7$$

$$H_{5,3} = \max \{H_{4,2} + s(x_5, y_3), \max_{k \geq 1} (H_{5-k,3}-W_k), \max_{l \geq 1} (H_{5,3-l}-W_l), 00\} = 0.7$$

$$H_{1,4} = \max \{H_{0,3} + s(x_1, y_4), \max_{k \geq 1} (H_{1-k,4}-W_k), \max_{l \geq 1} (H_{1,4-l}-W_l), 00\} = 0.0$$

$$H_{2,4} = \max \{H_{1,3} + s(x_2, y_4), \max_{k \geq 1} (H_{2-k,4} - W_k), \max_{l \geq 1} (H_{2,4-l} - W_l), 0\} = 0.4$$

$$H_{3,4} = \max \{H_{2,3} + s(x_3, y_4), \max_{k \geq 1} (H_{3-k,4} - W_k), \max_{l \geq 1} (H_{3,4-l} - W_l), 0\} = 0.7$$

$$H_{4,4} = \max \{H_{3,3} + s(x_4, y_4), \max_{k \geq 1} (H_{4-k,4} - W_k), \max_{l \geq 1} (H_{4,4-l} - W_l), 0\} = 1.4$$

$$H_{5,4} = \max \{H_{4,3} + s(x_5, y_4), \max_{k \geq 1} (H_{5-k,4} - W_k), \max_{l \geq 1} (H_{5,4-l} - W_l), 0\} = 2.7$$

Tabel hasil skor penyejajaran diatas adalah sebagai berikut:

**Tabel 4. Matriks H akhir**

	0	Saya	Kuliah	Teknik	Informatika
0	0	0	0	0	0
Saya	0	1.0	0.0	0.1	0.0
Kuliah	0	0.0	2.0	1.0	0.4
di	0	0.0	0.7	1.7	0.7
Teknik	0	0.0	1.0	1.7	4.0
Informatika	0	0.0	0.1	0.7	2.7

Nilai  $H_{i,j}$  yang memiliki nilai terbesar adalah  $H_{4,4}$  yaitu = 4.0

Setelah ditemukan nilai terbesar maka langkah selanjutnya adalah melakukan penelusuran kembali:

$$H_{5,4} \text{ didapat dari } H_{4,3} + s(x_4, y_4) = 2.7$$

$$H_{4,3} \text{ didapat dari } H_{3,2} + s(x_3, y_3) = 1.7$$

$$H_{3,2} \text{ didapat dari } H_{2,2} - W_1 = 0.7$$

$$H_{2,2} \text{ didapat dari } H_{1,1} + s(x_2, y_2) = 2.0$$

$$H_{1,1} \text{ didapat dari } H_{0,0} + s(x_1, y_1) = 1.0$$

Karena nilai  $H_{0,0} = 0.0$  maka prosedur dari penelusuran kembali telah selesai.

Jadi segmen yang memiliki tingkat kemiripan paling besar adalah:

Saya Kuliah - Teknik Informatika

Saya Kuliah di Teknik Informatika

Dengan penyejajaran yang memiliki nilai paling besar adalah :

Saya Kuliah - Teknik Informatika

Saya Kuliah di Teknik Informatika

Dengan nilai skor maksimal adalah 2.7

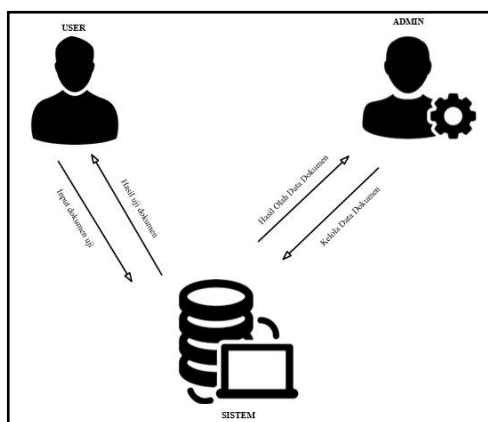
### Perhitungan Bobot

Dari hasil penelusuran yang sudah dilakukan dapat disimpulkan bahwa dari satu aligment tersebut terdapat 4 karakter yang sama dari total karakter 4 karakter dan aligment kedua terdapat 4 karakter yang sama dari total 5 karakter, maka hasilnya adalah:

$$\frac{((4/4) + (4/5))}{2} \times 100\% = 90\%$$



### 3.2. Perancangan Arsitektur Sistem



Gambar 1. Arsitektur Sistem

Perancangan arsitektur aplikasi dapat dijelaskan bahwa *user* menginputkan dokumen ke sistem kemudian sistem mengolah dokumen dengan proses *preprocessing* sebelum hasil uji dokumen dapat di lihat hasil persentase tingkat kemiripannya, kemudian data yang sudah diolah akan disimpan ke dalam data base dan hasil dari uji dokumen tersebut akan ditampilkan kepada *user*.

Dari sisi admin alur arsitekturnya yaitu admin menginputkan dan mengolah data dokumen ke sistem yang terdiri dari menginputkan data dokumen uji, memperbarui data dokumen uji, menambah data dokumen uji, dan melihat hasil uji dokumen yang akan menjadi data uji kemudian data dokumen tersebut akan disimpan ke dalam *database* dan hasilnya akan ditampilkan ke menu admin.

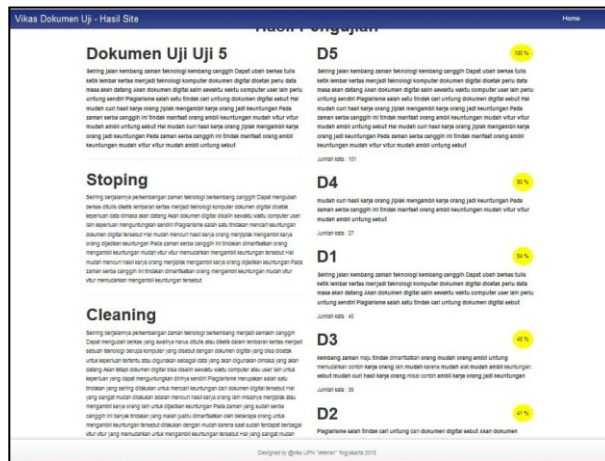
## 4. HASIL DAN PEMBAHASAN

Aplikasi pendeteksi plagiasi dokumen ini akan menghasilkan perbandingan uji kesamaan dokumen. Halaman uji dokumen dapat dilihat pada gambar 3.

The screenshot shows a web application interface for document upload. The page title is 'Masukkan dokumen uji'. Below the title, there is a text input field labeled 'Nama' with a note 'Fields with \* are required.'. Below the input field, there is a 'Browse...' button and a 'Create' button. The page also has a 'Home' link in the top right corner and a footer that reads 'Designed by @Vika UPI "releas" Yogyakarta 2015'.

Gambar 2. Tampilan halaman input dokumen uji

Pada gambar diatas merupakan tampilan halaman input dokumen, digunakan untuk menginputkan dokumen.



Gambar 3. Tampilan halaman input dokumen uji

Pada gambar diatas merupakan hasil perbandingan dokumen yang di uji dengan dokumen yang ada pada *database*. Hasil tersebut berupa persentase hasil kesamaan antar dokumen.

Tabel 5. Hasil Uji Coba Dokumen

No	Nama Dokumen	Isi Dokumen
1	D1	Seiring berjalannya perkembangan zaman, teknologi pun berkembang menjadi semakin canggih. Dapat mengubah berkas yang awalnya harus ditulis atau diketik ke dalam lembaran kertas menjadi sebuah teknologi berupa komputer yang disebut dengan dokumen digital yang bisa dicetak untuk keperluan tertentu atau digunakan sebagai data yang akan digunakan dimasa yang akan datang. Akan tetapi dokumen digital ini bisa disalin sewaktu-waktu ke computer atau user lain untuk keperluan yang dapat menguntungkan dirinya sendiri. Plagiarisme merupakan salah satu tindakan yang sering dilakukan untuk mencari keuntungan dari dokumen digital tersebut.
2	D2	Plagiarisme merupakan salah satu tindakan yang sering dilakukan untuk mencari keuntungan dari dokumen digital tersebut. Akan tetapi dokumen digital ini bisa disalin sewaktu-waktu ke computer atau user lain untuk keperluan yang dapat menguntungkan dirinya sendiri. Seiring berjalannya perkembangan zaman, teknologi pun berkembang menjadi semakin canggih. Dapat mengubah berkas yang awalnya harus ditulis atau diketik ke dalam lembaran kertas menjadi sebuah teknologi berupa computer yang disebut dengan dokumen digital yang bisa dicetak untuk keperluan tertentu atau digunakan sebagai data yang akan digunakan dimasa yang akan datang.
3	D3	Pada perkembangan zaman yang sudah semakin maju ini banyak tindakan yang malah justru dimanfaatkan oleh beberapa orang untuk begitu mudahnya orang mengambil keuntungan untuk memudahkannya dalam mencontoh karya orang lain. Hal tersebut dilakukan dengan mudah karena saat ini sudah terdapat berbagai kemudahan alat yang memudahkan untuk mengambil keuntungan tersebut. Hal yang sangat mudah dilakukan adalah mencuri hasil karya orang lain misalnya mencontoh atau mengambil karya orang lain untuk dijadikan keuntungan.
4	D4	Hal yang sangat mudah dilakukan adalah mencuri hasil karya orang lain misalnya menjiplak atau mengambil karya orang lain untuk dijadikan keuntungan. Pada zaman yang sudah serba canggih ini

		banyak tindakan yang malah justru dimanfaatkan oleh beberapa orang untuk mengambil keuntungan. Hal tersebut dilakukan dengan mudah karena saat ini sudah terdapat berbagai vitur-vitur yang memudahkan untuk mengambil keuntungan tersebut.
5	D5	Seiring berjalannya perkembangan zaman, teknologi pun berkembang menjadi semakin canggih. Dapat mengubah berkas yang awalnya harus ditulis atau diketik ke dalam lembaran kertas menjadi sebuah teknologi berupa komputer yang disebut dengan dokumen digital yang bisa dicetak untuk keperluan tertentu atau digunakan sebagai data yang akan digunakan dimasa yang akan datang. Akan tetapi dokumen digital ini bisa disalin sewaktu-waktu ke computer atau user lain untuk keperluan yang dapat menguntungkan dirinya sendiri. Plagiarisme merupakan salah satu tindakan yang sering dilakukan untuk mencari keuntungan dari dokumen digital tersebut. Hal yang sangat mudah dilakukan adalah mencuri hasil karya orang lain misalnya menjiplak atau mengambil karya orang lain untuk dijadikan keuntungan. Pada zaman yang sudah serba canggih ini banyak tindakan yang malah justru dimanfaatkan oleh beberapa orang untuk mengambil keuntungan. Hal tersebut dilakukan dengan mudah karena saat ini sudah terdapat berbagai vitur-vitur yang memudahkan untuk mengambil keuntungan tersebut.

Tabel diatas merupakan tabel dokumen pengujian dokumen 1 sampai dokumen 5 dengan menggunakan stemming. Tabel uji dokumen bertujuan untuk menganalisa persentase kemiripan antar dokumen. Berikut ini hasil persentase kemiripan antar dokumen dapat dilihat pada Tabel 6 dan Tabel 7.

**Tabel 6. Hasil Uji Coba Dokumen**

	D1 (%)	D2 (%)	D3 (%)	D4 (%)	D5 (%)
D1	100%	69%	8%	6%	45%
D2	60%	100%	8%	6%	32%
D3	5%	5%	100%	63%	23%
D4	4%	4%	41%	100%	27%
D5	54%	41%	45%	83%	100%

**Tabel 7. Hasil Uji Coba Dokumen**

N o	Nama Dokumen	Jumlah Kata
1	D1	45
2	D2	43
3	D3	35
4	D4	27
5	D5	101

Berdasarkan hasil pengujian dokumen diatas dapat dilihat persentase kemiripan antar dokumen. Pengujian dokumen ini menggunakan proses stemming yang berfungsi merubah isi dokumen ke bentuk kata dasar sehingga memudahkan perhitungan untuk menentukan hasil persentase dari kesamaan dokumen tersebut.

## 5. KESIMPULAN

Berdasarkan dari penelitian bahwa algoritma Smith-Waterman mampu mendeteksi kesamaan dokumen dengan menemukan letak kesamaan dari dokumen yang dibandingkan akan tetapi algoritma ini tidak dapat membandingkan dokumen dengan data yang banyak dan memiliki tingkat kearurasian yang rendah dan membutuhkan waktu yang lama, sehingga penulis menggunakan proses *preprocessing* untuk menentukan hasil yang lebih akurat dan mempercepat proses yaitu dengan menggunakan stemming Nazief & Andriani.

**DAFTAR PUSTAKA**

- Ariyanto, E. Y., Aman, M., & Rochmad, C. D. (n.d.). Perancangan Dan Pembuatan Sistem Penebar Pakan Ikan Jenis Pasta Otomatis Berbasis Mikrokontroler AT89S51, 2–4.
- Bangun, R., Pengkondisi, A., & Budidaya, K. (2017). Vol. 13, 13(3), 178–182.
- Cipto, G., Atmaja, T., Putrada, A. G., & Rakhmatsyah, A. (2018). Optimasi Tingkat Hidup Udang Crystal Red dengan Menerapkan Metode Fuzzy Logic Berbasis IOT, 5(2), 3649–3656.
- Design And Fabrication Of An Automatic Fish Feeding System For Home Aquarium Siti Aisyah Binti Saahri Report submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering ( Hons .) Mechatronics Engineering Faculty of Manufacturing Engineering Universiti Malaysia Pahang. (2015), (June).
- Dewantoro, W. (2016). Jurnal Ilmiah Komputer dan Informatika ( KOMPUTA ) Pembangunan Sistem Pantau Smart Fish Farm Menggunakan Arduino Berbasis Internet Of Things ( Iot ) Terhadap Budidaya Ikan Wisnu Dewantoro Teknik Informatika - Universitas Komputer Indonesia Jurnal Ilmiah Komputer dan Informatika ( KOMPUTA ), 2016, 1–8.
- E-journal, A., & Elektro, J. T. (2016). Rancang bangun perangkat pemberi pakan ikan otomatis pada kolam pembenihan ikan berbasis arduino.
- Irawan, D., Rosmiati, M., & Sularsa, A. (2017). Pembangunan Sistem Monitoring Penjadwalan Pemberian Makan Ikan Lele Berbasis Sms Gateway Development System Of Monitoring Schedule Feeding For Catfish, 3(3), 1833–1838.
- Ke-, P. S. (2017). C dan hasil kalibrasi dengan thermometer dihasilkan tingkat kesalahan penggunaan sensor DS18B20 adalah <2% Kata kunci : sensor, suhu, DS18B20, Arduino uno 1.
- Komaludin, D. (2018). Prototype Monitoring Suhu Tanaman Hidroponik Teknologi Iot ( Internet Of Thing ) Abstrak, 3(1), 45–51.
- Maulana, H., & Julianto, A. M. (2017). Pembangunan System Smartfishing Berbasis Internet of Things ( Studi Kasus di Peternakan Ikan Cahaya Ikan Mas , Majalaya ), 2017, 169–174.
- Mikrokontroller, M., & Uno, A. (2014). yang asalnya pemukiman penduduk kini hampir semua.
- Publikasi, N., Akhir, T., Waluyo, A., Studi, P., Elektro, T., Teknologi, F., ... Yogyakarta, U. T. (2018). Pemberi Pakan Ikan Otomatis Menggunakan ESP8266 Berbasis Internet Of Things ( IOT ) Pemberi Pakan Ikan Otomatis Menggunakan ESP8266 Berbasis Internet Of Things ( IOT ).
- Qalit, A., & Rahman, A. (2017). Rancang Bangun Prototipe Pemantauan Kadar pH dan Kontrol Suhu Serta Pemberian Pakan Otomatis pada Budidaya Ikan Lele Sangkuriang Berbasis IoT, 2(3), 8–15.
- Sarjana, P., & Ii, M. (2010). An Intelligent Fish Feeder System Nurul Husna Binti Abdul Razak This report in partial fulfillment of the requirements for the award of Bachelor of Electronic Engineering ( Industrial Electronics ) With Honours Faculty of Electronic and Computer Engineering Universiti Teknikal Malaysia Melaka April 2010, (April).
- Teknikal, U. (2007). Automatic Fish Feeder for Cultivation Pond, (April).
- Weku, H. S., Poekoel, E. V. C., Robot, R. F., & Eng, M. (2015). Rancang Bangun Alat Pemberi Pakan Ikan Otomatis Berbasis Mikrokontroler, 5(7), 54–64.
-