**LAPORAN TUGAS BESAR KECERDASAN BUATAN**

**(Plagiarism and Collusion Detection using the Smith-Waterman Algorithm)**

****

**Anggota:**

**10111931 Andrew Tooy**

**10110642 Maya Hermawati**

**10110649 Hengky Saputra**

**IF – 15**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK DAN ILMU KOMPUTER**

**UNIVERSITAS KOMPUTER INDONESIA**

**2013**

# PENDAHULUAN

Pendeteksian kemiripan dokumen merupakan pendeteksian kesamaan beberapa dokumen dengan membandingkan isi dokumen sehingga menghasilkan bobot atau nilai kemiripan dari dokumen yang dibandingkan. Salah satu kegunaan perbandingan isi dokumen adalah untuk membantu pengguna dalam pengelompokan dokumen dan juga memungkinkan pengguna mengetahui apakah isi dokumen yang satu merupakan dokumen yang pada dasarnya sama dengan dokumen yang lain. Hal ini berguna untuk mengetahui apakah sebuah dokumen merupakan hasil ciplakan dari dokumen lain.

Pendeteksian kemiripan dokumen ini dapat dilakukan dengan beberapa teknik, misalnya teknik pencarian informasi, teknik penghitungan statistik, atau dengan menggunakan informasi sintaktik dari kalimat perkalimatnya. Pendekatan-pendekatan tersebut tidaklah sempurna, masih terdapat beberapa kelemahan, misalnya penghitungan statistik yang membandingkan frekuensi kata dari dokumen satu dengan dokumen yang lain, tidak memperhatikan struktur kalimat. Sedangkan dalam teknik sintaktik kalimat, urutan kata dalam kalimat diperiksa unsur semantiknya dengan cara mengolah letak kata sesuai tatabahasanya atau dengan penggantian sebuah kata dengan sinonim dari kata tersebut. Teknik ini mempunyai kelemahan, yaitu setiap kata dikelompokkan pada label masing-masing untuk mengetahui struktur kalimat.

Perangkat lunak yang digunakan untuk mendeteksi kemiripan dokumen ini menggunakan algoritma Smith-Waterman. Algoritma ini pada awalnya dipakai untuk mengidentiﬁkasi (menemukenali) atau menghitung kemiripan isi pada penyejajaran urutan biologis. Biasanya algoritma ini digunakan dalam kasus pembandingan dua rangkaian DNA dengan melihat kesamaannya. Dengan penganalogian sebuah kata pada sebuah kalimat menjadi seperti sebuah gen atau protein DNA, maka algoritma ini dapat dipakai untuk membandingkan kesamaan kalimat.

Tahapan yang dilakukan untuk pendeteksian kemiripan dokumen teks adalah pembacaan dokumen teks, penghitungan, kesamaan struktur, pengoptimalan, stemming, dan penghitungan bobot dengan menggunakan algoritma smith-waterman. Hasil penelitian ini dapat dimanfaatkan untuk mendeteksi kemiripan sebuah dokumen dengan dokumen lain yang diduga sebagai hasil ciplakan atau plagiat.

## Plagiarisme

Plagiarisme selalu dipersepsikan sebagai penjiplakan yang fatal. Paling tidak, sebagian besar masyarakat kita berkeyakinan seperti itu. Dengan demikian, sang pelaku harus diberi sanksi yang tegas dan berat. Misalnya penurunan pangkat, pencabutan gelar, bahkan pemecatan. Keyakinan itu tidak salah, tetapi tidak sepenuhnya benar.

Definisi plagiarisme telah banyak dikemukakan para ahli. Nevile (2010) dalam *The Complete Guide Referencing and Voiding Plagiarism* mendefinisikan plagiarisme sebagai tindakan mengambil ide atau tulisan orang lain tanpa menyebutkan rujukan dan diklaim sebagai miliknya. Oleh karena itu, penulisan kutipan dan sumber menjadi indikator utama untuk menentukan seseorang melakukan plagiat atau tidak.

Pada praktiknya, plagiarisme dibedakan menjadi beberapa kategori. Sastroasmoro (2007) mengategorikan berdasarkan:

1. Aspek yang dijjiplak

Plagiarisme jenis ini dibagi empat kategori:

* Plagiarisme ide
* Plagiarisme isi
* Plagiarisme tulisan
* Plagiarisme total

Dari keempat kategori ini, kategori terakhirlah yang dianggap paling berat.

1. Proporsi yang dijiplak

Plagiarisme jenis ini dibedakan dalam tiga kategori, yaitu:

* Plagiarisme ringan (,30 persen)
* Plagiarisme sedang (30 persen-70persen)
* Plagiarisme berat (.70 persen)

Ada anggapan, jumlah kutipan menjadi penentu baik tidaknya suatu karya ilmiah. Semakin banyak kutipan, semakin baik. Padahal, jika jumllahnya tidak wajar bisa dianggap plagiat. Idealnya, proporsi ide atau gagasan penulis harus lebih dominant.

1. Pola plagiarisme

Plagiarisme jenis ini dibedakan menjadi dua kategori, yaitu:

* Plagiarisme kata demi kata (word for word)
* Plagiarisme mozaik (menggabungkan ide orisinil dengan ide orang lain).

Yang paling sering dilakukan dengan menyelipkan atau menggabungkan tulisan orang lain menjadi tulisan yang baru. Penulis pun tidak menyebutkan sumbernya sehingga seolah-olah tulisan itu miliknya.

1. Kesengajaan

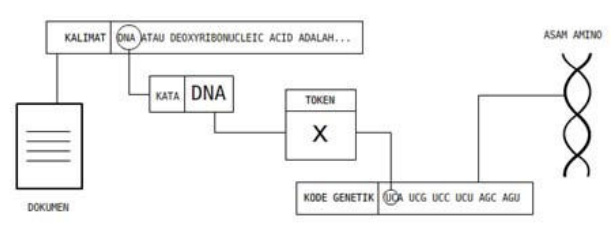
Plagiarisme jenis ini diklasifikasikan menjadi dua kategori, yaitu:

* Plagiarisme yang disengaja
* Plagiarisme yang tidak disengaja

Kategori kedua kerap terjadi dan menyebabkan seseorang dianggap plagiat. Misalnya, penulis lupa menuliskan sumber pada daftar pustaka. Padahal, di bagian isi, pengutipannya sudah benar. Walaupun terkesan remeh. Kelalaian ini bisa berakibat fatal. Hal ini pula yang terjadi pada salah satu dosen yang saat ini santer diberitakan karena diindikasikan melakukan plagiat

## Algoritma Smith-Waterman

Algoritma Smith-Waterman merupakan algoritma klasik yang telah dikenal luas dalam bidang bioinformatika yang dipakai sebagai sebagai metode yang dapat mengidentiﬁkasi penyejajaran sekuens yaitu proses penyusunan dua rangkaian/susunan atau rentetan nukleotida atau susunan protein sehingga kemiripan antara dua rangkaian tersebut akan terlihat.



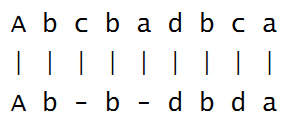
Berdasarkan fungsi proses penyejajaran sekuens tersebut, algoritma ini dapat diterapkan dalam pemprograman komputer untuk pendeteksian kesamaan atau pengukuran tingkat kemiripan sebuah dokumen teks dengan dokumen teks lain dengan cara melihat kesamaan isi (local similarities) dari kedua dokumen teks tersebut.

Dalam gambar 1 diandaikan bahwa lambang X dan Y merupakan dua urutan (sekuens) string yang masing-masing berasal dari dokumen yang berbeda. Panjang X dan Y masing-masing dinyatakan sebagai m dan n. Dari dua string ini dapat dihitung nilai kecocokan yang diperoleh dari pembandingan substring X dari string X dengan substring Y dari string Y.

Proses pembandingan ini akan menghasilkan penyejajaran yang identik/mirip (hit) dengan atau tanpa perubahan urutan string seperti penghilangan (deletion), penyisipan (insertion), dan penggantian (replacement). Anggap h adalah kontribusi positif yang merepresentasikan hit atau cocok, d untuk kontribusi negatif yang merepresentasikan penyisipan atau penghilangan (atau bisa disebut indel), sedangkan r adalah kontribusi negatif yang dibuat dengan menggantikan satu simbol dengan simbol yang lain.

Model yang lebih umum pada khususnya digunakan di dalam bidang biologi komputasional. Dengan menggunakan nilai positif untuk identik dan nilai negatif untuk penghilangan dan penggantian, maka pembentukan nilai dari tiap-tiap simbol dari dua buah string tersebut dapat direpresentasikan di dalam bentuk matriks. Namun demikian, nilai hubungan antara h, d, dan r belum diketahui secara jelas.

Pada intinya, dalam pengidentiﬁkasian kesamaan string digunakan prinsip penambahan dan pengurangan. Huruf h merepresentasikan penambahan, sedangkan d dan r merepresentasikan pengurangan. Dapat diasumsikan bahwa penambahan dan pengurangan memiliki bobot yang sama. Dengan demikian dapat dianggap nilai dari h, d, dan r adalah 1. Sebagai contoh, bila sub-string X = abcbadbca dan substring Y = abbdbda, dengan penyejajaran yang optimal didapatkan 6 hit, 2 indel, dan 1 replacement, seperti yang ditunjukkan pada gambar di bawah ini, dan didapatkan nilai untuk dua string yang diberikan ini, yaitu 6h - 2d - r, atau 6 - 2 - 1 = 3 untuk kasus h = d = r = 1 dengan keterangan tanda “|” menunjukan kecocokan atau match, sedangkan tanda “-“ menunjukan adanya kesenjangan atau gap di antara dua sekuens string.



Tujuan dari pembandingan substring ini yaitu menemukan kemiripan yang *significant* sesuai dengan nilai ambang batas (*threshold*). Bila nilai hasil pembandingan di bawah nilai ambang batas, maka dianggap kemiripannya tidak *significant* dan bila nilainya di atas atau sama dengan nilai ambang batas, maka dianggap kemiripannya *significant*.

Algoritma standar Smith-Waterman yang dipakai untuk perhitungan *local alignment* adalah:

1. Gunakan rumus ini untuk menentukan nilai matriks dan symbol traceback



Dimana:

* = string sepanjang alphabet 
* = panjang
* = panjang
*  = maksimum skor kesamaan diantara suatu suffix dari a[1…i] dan suatu suffix dari b[1…j]

, ‘-‘ adalah skema gap-scoring

1. Inisialisasi awal matriks dengan nilai 0 (nol)
2. Inisialisasi nilai pembanding untuk character yang sama maupun yang tidak sama serta nilai gap-scoring
3. Mencari nilai matriksnya
4. Semua nilai yang terdapat dalam matriks apabila lebih kecil dari 0 (nol), maka nilai dianggap 0 (nol)
5. Memulai *traceback* dari nilai yang tertinggi yang ditemukan dimanapun pada matriks

Untuk melakukan *traceback*, bisa menggunakan simbol panah dengan berdasarkan nilai matriks yang didapatkan:

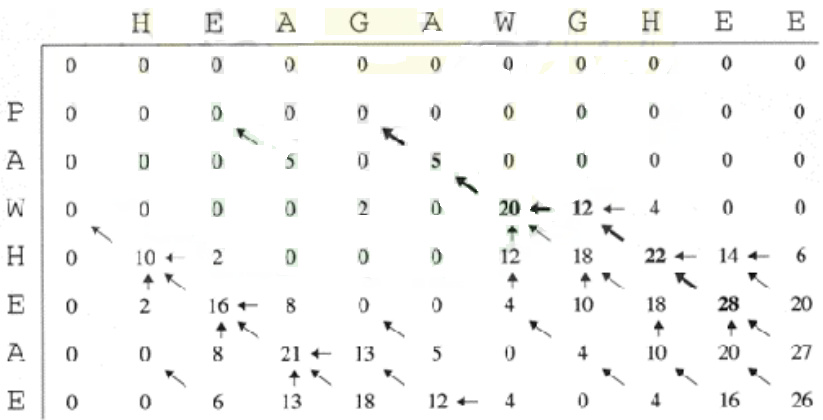
 untuk Match/Mismatch

 untuk Insertion

 untuk Deletion

1. Traceback dilanjutkan hingga skor diposisinya bernilai 0 (nol)

Setelah didapat urutan traceback, kemudian hitung banyaknya symbol *match/mismatch*. Lalu jumlah *match/mismatch* tersebut dibagi dengan *length* dari string terpanjang dan hasilnya dikalikan dengan 100% untuk mendapat persentase plagiat-nya yang nantinya akan dibandingkan dengan kategori proporsi yang dijiplak.



Menurut Robert W. Irving skema pemrograman dinamis algoritma klasik Smith-Waterman dibagi menjadi dua bagian yaitu:

1. Didefinisikan Sij menjadi nilai maksimum yang didapatkan dari proses perbandingan sebuah substring A pada posisi ke-i dengan sebuah substring B pada posisi ke-j. hubungan rekurens standar untuk Sij, yaitu:

* Jika A(i) = B(j) maka Sij = Si-j, j-1+h, atau
* Jika A(i) ≠ B(j) maka Sij = max(0, Si-1, j-d, Si, j-1-d, Si-1, j-1-r)
* Dimana kondisi awal adalah Si,0 = S0, j = 0 untuk semua I,j

1. Digunakan ide *traceback path* untuk mengkontruksi sebuah *local alignment* yang optimal pada posisi ke-i substring A dan posisi ke-j substring B agar lebih jelas terlihat. Dengan diberikan sel (i,j), dapat didefinisikan sebuah sel *parent* sebagai berikut:

* Jika Sij = 0, maka sel (i,j) tidak mempunyai parent
* Jika A(i) = B(j), maka sel (i,j) mempunyai parent sel (i-1, j-1)
* Sebagai tambahan, sel (i,j) mempunyai parent yaitu untuk sel (p,q) ϵ {(i-1,j),(I,j-1)} sehingga Sij = Spq-d, dan/atau sel (i-1,j-1) jika Sij = Si-1, j-1-r

Jadi, setiap sel yang mengandung nilai tidak nol mempunyai paling tidak sedikitnya satu *parent*, dan juga mungkin saja dapat memiliki tiga buah *parent*.

## Pengolahan Awal

Unsur-unsur yang digunakan untuk mengidentiﬁkasi kemiripan teks diimplementasikan ke dalam pemprograman denggan menggunakan algoritma standar Smith-Waterman dengan fungsi penyejajaran sekuensnya. Akan tetapi, algoritma standar tersebut memiliki sifat yang hanya membandingkan secara eksplisit dua string tanpa mengetahui sifat karakter-karakter yang membentuk kedua string tersebut.

Oleh karena itu dibutuhkan proses bantuan (pre-processing) dalam bentuk modul tambahan, modul tersebut tersebar dalam pembobotan sesuai fungsi modul. Pengolahan awal ini akan menambah waktu proses sistem secara keseluruhan. Dengan adanya pengurangan noise yang dilakukan pengolahan awal ini dapat mengurangi kompleksitas pada saat pembandingan oleh algoritma Smith-Waterman.

## Stemming

Stemming merupakan suatu proses untuk menemukan kata dasar dari sebuah kata dengan cara penghilangan imbuhan (aﬃxes) yang berupa awalan (preﬁxes), sisipan (inﬁxes), akhiran (suﬃxes), atau conﬁxes (awalan dan akhiran) pada kata turunan. Misalnya kata berlari dan dilarikan akan diubah menjadi sebuah kata yang sama yaitu lari. Metode stemming memerlukan masukan berupa kata yang terdapat dalam suatu dokumen, dengan menghasilkan keluaran berupa kata dasar.

Pencarian kata dasar (akar kata) dari suatu kata yang berimbuhan dalam bahasa Indonesia merupakan pekerjaan yang kompleks. Berbeda dengan bahasa Inggris yang hanya memiliki imbuhan berupa sufﬁxes (akhiran), kata-kata dalam bahasa Indonesia bisa memiliki imbuhan yang terdiri atas preﬁxes (awalan), suﬃxes (akhiran), inﬁxes (sisipan) dan conﬁxes (kombinasi dari awalan, akhiran dan sisipan).

## Karakteristik Struktural

Karakteristik Struktural adalah struktur pengindeksan internal dari tiap dokumen yang dihasilkan dengan menggabungkan struktur pohon dokumen beserta himpunan kata kuncinya. Dalam pendeteksian kemiripan dokumen, karakteristik struktural atau disebut kesamaan struktural, digunakan untuk menguji dua buah dokumen yang akan dibandingkan lebih jauh lagi karena adanya persamaan kata kunci yang terkandung dalam kedua dokumen tersebut. Hal ini bisa dilihat dari himpunan kata kunci yang unik.

Bila kedua dokumen memiliki himpunan kata kunci yang relatif berbeda, maka dianggap bahwa kedua dokumen tersebut memiliki kandungan yang berbeda. Sebaliknya, bila kedua dokumen memiliki himpunan kata kunci yang relatif sama, maka dapat dianggap bahwa salah satu dokumen mungkin saja sama terhadap dokumen yang lain. Proses selanjutnya adalah pembandingan kalimat per kalimat.

Dalam proses ini dilakukan proses pembandingan struktural level dua. Bila dua kalimat memiliki himpunan kata kunci yang relatif berbeda, maka diasumsikan bahwa kedua kalimat tersebut memiliki makna semantik yang berbeda, demikian sebaliknya. Dengan cara ini, pembandingan dengan algoritma Smith-Waterman hanya dilakukan bila keadaan struktural kedua kalimat dalam kedua dokumen memiliki kesamaan atau kemiripan. Makna kemiripan ditentukan melalui deﬁnisi nilai ambang (threshold) yang dianggap signiﬁkan untuk mendeteksi kemiripan dokumen.

## Proses Tambahan

Proses tambahan bertujuan menghilangkan noise pada dokumen untuk mempermudah pembandingan (memperkecil kompleksitas) pada saat pembobotan oleh algoritma Smith-Waterman. Penghilangan noise terdiri atas:

1. Pengubahan penulisan angka secara numerik menjadi penulisan angka secara alphabet. Proses ini bersifat mempermudah pembadingan kata dengan menyeragamkan penulisan angka tanpa pengubahan makna kalimat.
2. Penghilangan kata yang tergolong closed-class word dan stop words, closed-class word adalah kelas kata yang keanggotaannya biasanya kecil dan stabil. Kebalikan dari closed-class word adalah open-class word, yaitu kelompok kata yang keanggotaannya biasanya besar dan senantiasa bertambah, hal ini bisa dikarenakan faktor serapan bahasa lain, teknologi baru ataupun adanya imbuhan. Yang termasuk di dalam closed-class word yaitu preposisi, penentu, kata ganti, konjungsi, kata bantu, partikel dan angka.

Oleh karena closed-class word merupakan bahasa yang jumlahnya terbatas maka dapat dilakukan proses eliminasi pada pendeteksian kemiripan dokumen, tetapi tidak semua kelas kata dihilangkan karena ada beberapa kelas kata yang mempunyai nilai semantik yang dapat mengubah makna dari suatu kalimat atapun memberikan efek ambigu. Kelas kata yang tidak akan dihilangkan dari proses eliminasi yaitu kelas kata penentu, angka, partikel, kata ganti dan sebagian preposisi. Stop words adalah kata umum (common words) yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna. Stop words umumnya dimanfaatkan dalam pencarian informasi, stop words untuk bahasa Inggris antara lain of, the, sedangkan untuk bahasa Indonesia antara lain yang, di, dan ke.

# PEMBAHASAN

Algoritma Smith-Waterman secara matematis dapat menemukan *local alignment* (*highest-scoring*) yang terbaik dari dua buah sekuens string. *Best local alignment* adalah *alignment* terbaik dari semua kemungkinan substring (bagian) dari string pertama dan kedua. Proses pencarian *local alignment* bertujuan untuk menemukan tingkat kesamaan yang tinggi dari substring tertentu antara sepasang rangkaian dengan mengabaikan ketidaksamaan rangaian secara keseluruhan.

Perhitungan *local alignment* ini dapat dihitung dengan menggunakan sebuah matriks T. Matriks T dibangun sebagai berikut:

T(*i,*0) = 0, 0 ≤ *i* ≤ *m*

T(0,*j*) = 0, 0 ≤ *j* ≤ *n*

Jika  maka (match) atau jika  maka (mismatch)



Dimana:

* = string sepanjang alphabet 
* = panjang
* = panjang
*  = maksimum skor kesamaan diantara suatu suffix dari a[1…i] dan suatu suffix dari b[1…j]
* , ‘-‘ adalah skema gap-scoring

Misalkan terdapat dua string yang akan dibandingkan yaitu:

String 1 = ACACACTA

String 2 = AGCACACA

Perbandingan untuk kedua string diatas dengan menggunakan algoritma Smith-Waterman adalah sebagai berikut:

1. **Tambahkan sebuah nilai pada setiap pembanding**

 = +2

 = -1

1. **Inisialisasi awal matriks T dengan nilai 0 (nol)**

****

1. **Hitung , jika nilai lebih kecil dari 0 (nol) maka nilai dianggap 0 (nol)**

Pertama hitung  cara sebagai berikut:



Nilai maksimum adalah 2, maka  = 2

Selanjutnya hitung , … dengan menggunakan cara yang sama.



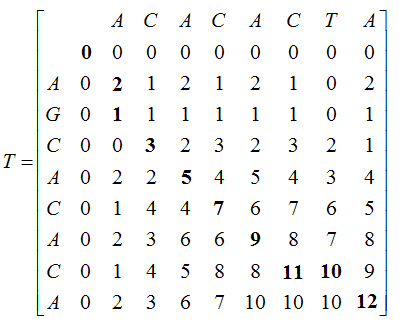
1. **Lakukan *traceback* dari nilai yang tertinggi yang ada pada matriks**

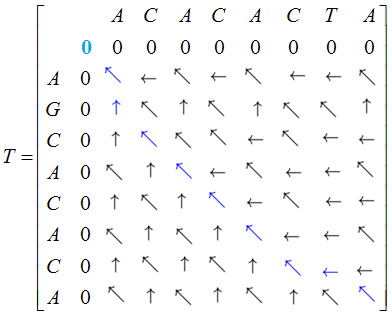
Untuk memudahkan dalam melakukan *traceback*, bisa menggunakan simbol panah dengan ketentuan:

 : Match/Mismatch

 : Insertion

 : Deletion

****

****

Nilai tertinggi pada matriks adalah 12 maka *traceback* dilakukan dari nilai 12 () dan ditelusuri ke atas atau ke kiri selama nilai masih positif. Urutan *traceback* untuk matriks diatas adalah:

(8,8), (7,7), (7,6), (6,5), (5,4), (4,3), (3,2), (2,1), (1,1), dan (0,0)

Tingkat kesamaan diantara dua buah string dapat dihitung dengan cara:

Jumlah symbol match/mismatch dari urutan traceback = 7

Jumlah string terpanjang = 8

Persentase = 7/8 \*100% = 87.5%

Persentase tersebut kemudian dibandingkan dengan kategori plagianya, dan hasilnya :

87.5% tergolong plagiat berat.

# PENUTUP

## Kesimpulan

1. Plagiarism atau collision bisa dideteksi dengan menggunakan algoritma Smith-Waterman.
2. Algoritma Smith-Waterman awalnya digunakan dalam area biologi molekul, dan sekarang bisa dikembangkan ke area pencarian perbandingan kesamaan suatu teks dokumen (atau computer program)