JURNAL ILMU KOMPUTER DAN DESAIN KOMUNIKASI VISUAL (JIKDISKOMVIS)
Volume 2 No. 1 Desember 2017
ISSN: 2541-4585

Perbandingan Penggunaan Algoritma Cosinus dan Wu Palmer untuk Mencari Kemiripan Kata dalam *Plagiarism Checker*

[1]Aslihatul Millah, [2]Siti Nurazizah
[1,2] Program Studi Sistem Informasi, Fakultas Sains dan Teknologi,
UIN Sunan Ampel Surabaya,
Jalan Ahmad Yani No. 117, Jemur Wonosari, Wonocolo, Kota Surabaya, Jawa Timur
[1]azzaalmillah@gmail.com [2]azaizah8@gmail.com

Abstrak

Plagiasi merupakan hal yang sangat penting ditekan dan dihindari,khususnya di lingkungan akademisi. Seringkali plagiasi ini terjadi baik dengan disengaja maupun tidak disengaja, namun hal ini dapat diminimalisisr dengan menggunakan plagiarism checker. Tujuan makalah ini adalah untuk perbandingan penggunaan algoritma cosinus dan wu palmer dalam mendeteksi plagiat berdasarkan kemiripan kata. Metode yang digunakan adalah menggunakan algoritma cosinus dan wu palmer yang kemudian dilakukan pengujian terhadap dua algoritma tersebur. Makalah ini mendapatkan hasil bahwa Cosinus bisa digunakan sebagai algoritma dalam mendeteksi plagiasi berdasarkan kemiripan kata.Jika hasil nilai perhitungan cosinus 0 – 0,5 maka tidak plagiat, jika > 0,5 – 1 maka plagiasi. Algoritma cosinus lebih efektif digunakan untuk mendeteksi plagiasi daripada algoritma wu palmer.

Kata Kunci : *Cosinus, Cosine Similarity, Wu Palmer*

1. PENDAHULUAN

1.1 Latar Belakang

Pada dasarnya beberapa ide dan karya yang kita tuliskan tidak sepenuhnya murni dari pemikiran kita. Jelas wajar bagi kita karena memang begitulah fitrah manusia,terbatas pengetahuannya. Hal ini dapat disikapi dengan beberapa Teknik seperti sitasi dan pengutipan yang benar. Namun,sering kali kita lupa menyebutkan sumber darimana kita mengutip ide atau karya tersebut atau mengutip dengan cara yang salah sehingga mengakibatkan karya yang kita hasilkan berpotensi terdeteksi sebagai plagiat.

Dewasa ini banyak kasus yang terjadi seputar plagiat seperti berita yang ditayangkan dalam salah satu situs berita bahwa demi mengejar gelar guru besar, 3 dosen dari salah satu universitas di Indonesia yang tidak dapat disebutkan identitasnya nekat melakukan plagiat untuk melancarkan aksinya (Anang, K.,

2017). Sanksi di perguruan tinggi pun juga mulai disemarakkan bahwa ada sanksi tegas pada akademisi yang nekat melakukan *plagiarism*e dalam menunaikan Tri Dharma Perguruan Tinggi mereka.

Untuk berhati-hati dan menjauhkan dari hal-hal yang tidak diinginkan terkait *plagiarism*, beberapa cara bisa ditempuh seperti melakukan Teknik sitasi yang baik atau menggunakan *tools* seperti *plagiarism checker.* Algoritma dan metode yang diterapkan antar *tools* pun juga berbeda-beda. Dalam sebuah jurnal karya Radiant Victor,dkk memaparkan bahwa algoritma cosinus dapat diimplementasikan bersamaan dengan algoritma swith waterman dalam mendeteksi kemiripan teks (Radiant Victor Imbar,dkk, 2014). Makalah ini membahas perbandingan penggunaan algoritma cosinus dan wu palmer dalam mendeteksi plagiat berdasarkan kemiripan kata. Makalah ini diharap mampu memberikan pernyataan yang signifikan algoritma mana diantara keduanya yang lebih efektif digunakan sebagai algoritma penyusun sistem *plagiarism checker*. Karena sering kali kita menyalin tulisan seseorang kemudian kita mengganti struktur kalimatnya saja ataupun tanpa menyebutkan sumbernya.

1.2 Identifikasi Masalah

Menentukan apakah kalimat tersebut plagiat atau tidak dengan menggunakan perhitungan cosinus berdasarkan kemiripan kata. Semakin besar nilai kemiripannya maka semakin berpotensi mengandung plagiat.

2. TINJAUAN PUSTAKA

2.1. Plagiat

Plagiat merupakan pengambilan karangan (pendapat dan sebagainya) orang lain dan menjadikannya seolah-olah karangan (pendapat dan sebagainya) sendiri, misalnya menerbitkan karya tulis orang lain atas nama dirinya sendiri; jiplakan (Anang, K., 2017). Sudah seharusnya para peneliti dan akademisi menghindari hal ini, dan bahkan plagiasi merupakan sebuah kejahatan dalam dunia akademis (KBBI).

Berikut adalah contoh beberapa implementasi yang sering dilakukan dan dihitung sebagai plagiat (Radiant Victor Imbar,dkk, 2014) :

2.1.1. Menyebut karya dan ide milik orang lain sebagai miliknya atau menyalin tulisan tanpa memberikan kredit pada penulis atau bahkan tanpa menulis sumbernya

2.1.2. Mengutip ide orang lain tanpa memberikan tanda kutip maupun tanda sitasi

2.1.3. Mengutip dengan cara yang salah

2.1.4. Membuat ide serupa hanya dengan merubah struktur kalimat tanpa penambahan variabel apapun dan tidak mencantumkan sumber.

2.2. Cosinus

Metode cosinus merupakan metode yang digunakan untuk menghitung *similarity* (tingkat kesamaan) antar dua buah objek (Na'firul Hasna Ariyani,dkk., 2016). Cosinus adalah metode untuk pengukuran kesamaan antara kata. Cosinus masih belum bisa menangani makna semantik teks dengan sempurna (Ogie Nurdiana, dkk., 2016). Dengan cosinus ini, sebuah kata bisa diteliti tingkat kemiripannya, mirip dalam artian ini bukan berarti sama. Cosinus memiliki *range* nilai antara 0-1.

2.3. Text Mining

Text mining adalah mencari dan mengukur data yang berupa teks dimana sumber data biasanya di dapatkan dari dokumen, dan tujuannya adalah mencari kata-kata yang dapat merepresentasikan isi dan maksud dari sebuah dokumen untuk kemudian dilakukan Analisa lebih lanjut (Departemen Pendidikan Nasional, 2005). Analisa yang dimaksud seperti mencari makna *semantic*, kemiripan kata, dan lainnya. *Text mining* juga bisa dianggap sebagai penerapan dari konsep data mining dalam mencari pola di dalam teks. Pola yang dimaksud adalah mencari intisari dari kata yang mewakili isi dari dokumen tersebut untuk mendapatkan informasi.

2.4. Metode *Term Frequency and Inverse Document Frequency* (TF IDF)

TF adalah *term frequency* dan IDF adalah *inverse term frequency.* Metode ini merupakan metode untuk menghitung bobot setiap setiap kata yang paling umum digunakan dalam multi disiplin ilmu *natural language processing* (Ogie Nurdiana, dkk., 2016). Dengan menggunakan TF IDF ini dapat diketahui bobot dari setiap kata/*term.* Dengan menggunakan TF IDF ini dapat diketahui bobot dari setiap kata/*term.* TF IDF ini merupakan suatu algoritma yang paling umum digunakan dalam *information retrieval.*

2.5. Wu Palmer

Wu palmer adalah sebuah algoritma path based. Wu palmer juga biasa disebut dengan algoritma Wu and Palmer (WUP). Algoritma ini adalah algoritma kemiripan semantik sehingga mampu mengukur derajat keterkaitan atau relevansi antar dokumen ataupun antar *term* (Paratisa Kharismadita dan Faisal Rahutomo, 2015). Algoritma ini bisa memberikan rekomendasi pada kasus-kasus yang membutuhkan pemeringkatan makna semantik berdasarkan kemiripan makna *semantic* tersebut.

2.6. Penelitian Terdahulu

Sebuah penelitian yang dilakukan oleh Paratisa Kharismadita dan Faisal Rahutomo dengan judul "*Implementasi Tokenizing Plus Pada Sistem Pendeteksi Kemiripan Jurnal Skripsi*" membahas tentang pendeteksi kemiripan jurnal skripsi untuk mengetahui apakah sebuah jurnal dapat dikatakan *plagiarism*e atau penjiplakan. Metode yang digunakan yaitu dengan menghitung 2 dokumen dengan menggunakan metode *Term Frequency and Inverse Document Frequency* (TFIDF) sebagai perhitungan term disetiap dokumennya dan *Cosine Similarity* untuk menghitung kemiripan antara 2 jurnal yang menghasilkan nilai 0 jika kedua jurnal sangatlah berbeda dan nilai 1 jika kedua jurnal mempunyai term yang sama (Ogie Nurdiana, dkk., 2016).

Menurut Hanto dan Harianto Kristanto dalam jurnalnya yang berjudul "*Program Bantu Pemilihan Lagu Pujian Berdasarkan Tema Kebaktian Dengan Menggunakan Metode Cosinus Similarity Studi Kasus: GKI Ngupasan*" bahwa

untuk memutuskan penggunaan lagu pujian dalam suatu kebaktian bukanlah perkara yang mudah. Peneliti membuat sebuah sistem yang menghasilkan suatu informasi berupa nilai perhitungan dari *cosine similarity* dengan bentuk persentase yang nantinya dapat menjadi acuan dalam memilih lagu yang sesuai dengan tema kebaktian. Dengan demikian penggunaan metode ini akan memperhitungkan antara kata dari tema kebaktian dengan kata pada lirik dari lagu (Akip Maulana, dkk, 2016).

3. METODE PENELITIAN

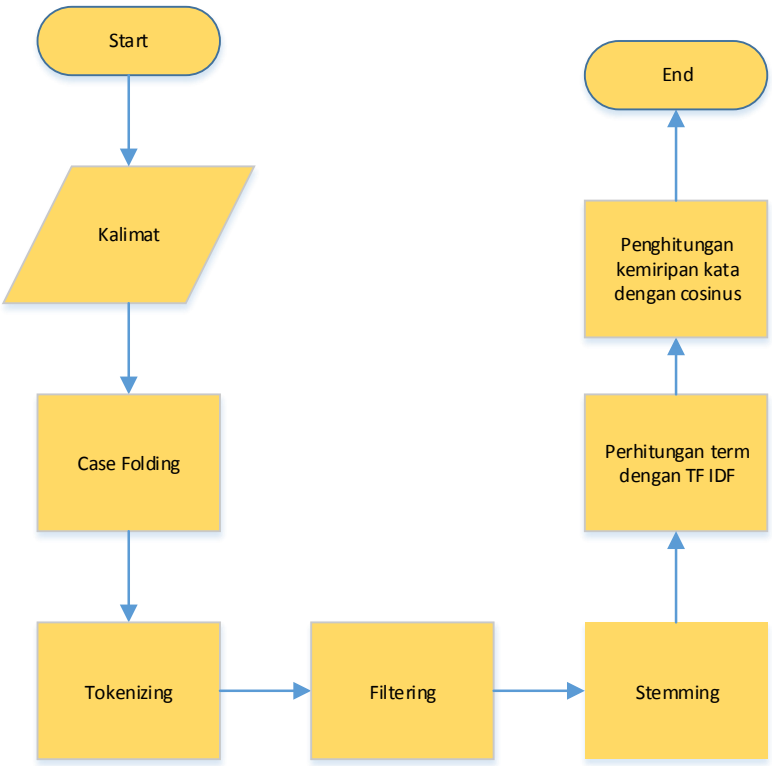
3.1 Rancangan Penelitian

Penelitian ini menggunakan metode cosinus untuk menguji plagiat bedasarkan kemiripan kata. Algoritma cosinus hanya dapat memberikan nilai *range* antara 0-1. Berikut adalah langkah-langkah untuk mendeteksi plagiat dengan menggunakan algoritma cosinus :

Misal :

teks sumber : Peneliti membuat program dengan bahasa java

teks target : Peneliti melakukan *coding* dengan bahasa java



Gambar 1. Alur Pengujian

Kalimat yang dimaksud di sini adalah karya tulis ilmiah maupun abstrak yang akan dideteksi plagiasinya. Namun,dalam perhitungan ini peneliti membahasi dengan menggunakan teks sumber dan teks target dengan kalimat yang telah disebutkan di atas.

1. Tahap *Case Folding*

Membuat semua teks menjadi seragam huruf kecil, dan hanya menerima huruf A-Z saja. Contoh : Peneliti membuat program dengan bahasa java menjadi "peneliti membuat program dengan bahasa java".

2. Tahap *Tokenizing*

Memisahkan seluruh tiap-tiap teks penyusunan komponen dokumen.

Contoh : Peneliti membuat program dengan bahasa java menjadi peneliti, membuat, program, dengan, bahasa, java.

3. *Filtering*

Memisahkan kata yang penting dari kata yang tidak penting dalam teks tersebut. Maksudnya adalah memisahkan teks dari kata yang dinilai tidak layak menjadi pembeda atau menunjukkan isi dari teks. Contoh : Peneliti membuat program dengan bahasa java menjadi peneliti, membuat, program, bahasa, java.

4. *Stemming*

Stemming adalah proses mengubah kata yang sudah di *filter* menjadi bentuk kata dasarnya. Langkah ini dinilai sangat membantu mengemukakan deteksi plagiasi. *Stem* (akar kata) adalah bagian dari kata yang tersisa setelah dihilangkan imbuhanya (awalan atau akhiran).

Contoh : peneliti, membuat, program, bahasa, java menjadi teliti, buat, program, java.

Setelah melalui proses di atas,kalimat siap dianalisis lebih lanjut untuk mengetahui apakah kalimat ini mengandung kemiripan kata yang plagiat atau tidak.

4. HASIL DAN PEMBAHASAN

4.1. Perhitungan TF IDF

Perhitungan ini adalah salah satu cara yang digunakan untuk menghitung kata dalam satu atau semua dokumen. Dengan menggunakan rumus sebagai berikut :

$$tf = tf$$

$$idf = log\frac{N}{df}$$

$$W = tf.idf$$

Keterangan:

tf : *term frequency*

idf : *inverse document frequency*

W : bobot kalimat terhadap kata

4.2. Perhitungan *Cosinus Similarity*

Perhitungan ini digunakan untuk menghitung kemiripan kata pada kalimat. Berikut rumus perhitungan *Cosinus Similarity*:

$$\text{Cos t1.t2} = \frac{t1.t2}{||t1||.||t2||}$$

Keterangan:

Cos t1.t2 : nilai kemiripan antarakalimat ke - 1 dan kalimat ke - 2

t1.t2 : jumlah kata dalam kalimat ke - 1 dan kalimat ke - 2

$||t1||.||t2||$: total kata dalam kalimat ke - 1 dan kalimat ke - 2

Untuk menghitung kalimat yang memiliki kemiripan kata, maka perlu memiliki 2 kalimat yaitu kalimat ke - 1 dan kalimat ke - 2. Dengan ketentuan kalimat sebagai berikut:

4.1. Kalimat ke - 1 : Peneliti membuat program dengan bahasa java

4.2. Kalimat ke - 2 : Peneliti melakukan *coding* dengan bahasa java

	peneliti	membuat	program	dengan	bahasa	java	melakukan	*coding*
t1	1	1	1	1	1	1	0	0
t2	1	0	0	1	1	1	1	1

$$\text{Cos t1 x t2} = \frac{t1 \text{ x } t2}{||t1|| \text{ x } ||t2||}$$

$$\text{Cos t1 x t2} = \frac{1x1 + 1x0 + 1x0 + 1x1 + 1x1 + 1x1 + 0x1 + 0x1}{\sqrt{1^2 \text{x } 1^2}x1^2x1^2x1^2x1^2 \text{x } 0^2x0^2x\sqrt{1^2 \text{x } 0^2}x0^2x1^2x1^2x1^2 \text{x } 1^2x1^2}$$

$$\text{Cos t1 x t2} = \frac{4}{\sqrt{6} \text{ x} \sqrt{6}}$$

Cos t1 x t2 = 0,666666667

Jadi kemiripan kata pada 2 kalimat tersebut adalah sebesar 0,666666667yang artinya kalimat tersebut plagiat. Perhitungan cosinus telah disebutkan di atas untuk mengetahui kemiripan kata di tiap dokumen untuk mendeteksi plagiasi, namun sayangnya cosinus ini hanya mampu mendeteksi dari segi kemiripan kata. Wu palmer mampu mengukur derajat kemiripan makna semantik antar kata.

$$\text{Score} = 2 * \text{depth(lcs)}/(\text{depth(s1)} + \text{depth(s2)})$$

Depth s1 merupakan kedalaman dari kata ke pertama dalam *wordnet* (leksikal database) yang berisi banyak *dataset* atau disebut *ontology*, *depht* s2 juga begitu. *Score* yang dihasilkan dalam rentang nilai 0 sampai 1 ($0 \leq score \leq 1$) [9].Wu palmer ini memiliki kinerja dengan proses perhitungan mencari jalur terpendek dari setiap *concept*, kemudian setiap jalur yang terbentuk digabungkan untuk mencari lcs-nya. Pencarian LCS (*Lowest Common Subsumer*) dengan cara mencari *sense* yang sering dimuncul dari dua jalur yang dihubungkan (Akip Maulana, dkk, 2016). Wu palmer akan mencari kata dengan makna *semantic* yang terkait baik dari segi sinonim, hipernim dan akronimnya.

4.3. Validasi *Range* Nilai

Perhitungan TF IDF yang dilanjutkan dengan perhitungan cosinus akan menghasilkan nilai kemiripan antar teks. Jika kedua dokumen bernilai 1 maka dokumen tersebut sama/plagiat (Paratisa Kharismadita dan Faisal Rahutomo, 2015). Peneliti mengadopsi sistem *rule base* yang terdapat *fuzzy* untuk mengklarifikasikan batasan nilai/*range* nilai mendeteksi plagiat atau tidak. Jika semakin ke angka 1 maka semakin plagiat dan sebaliknya, maka peneliti menyimpulkan setengah/separuh dari jarak 0 – 1 yang merupakan nilai yang dihasilkan oleh cosinus. Maka *range* nilai yang di dapat adalah :

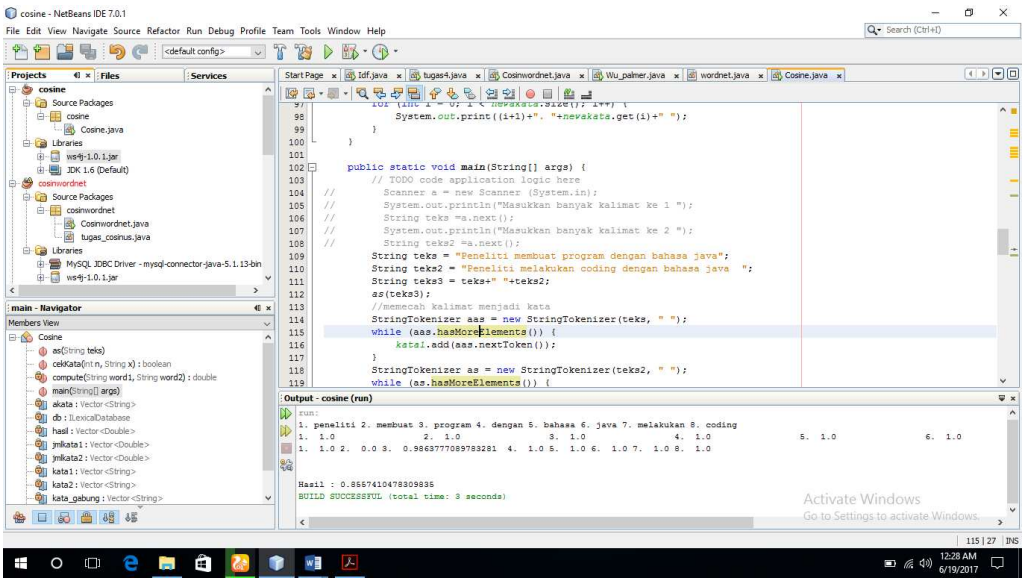
a. Jika nilai 0 – 0,5 maka tidak plagiat

b. Jika > 0,5 – 1 maka plagiat

Sedangkan untuk algoritma wu palmer, perhitungan tidak akan selesai jika tidak ada *dataset* sebelumnya. *Data set* ini berfungsi sebagai *word net* atau leksikal database. Hal ini ibarat kamus *thesaurus* yang dapat manafsirkan makna semantik atau makna terkait.

4.4. Pengujian

Pengujian *plagiarism checker* menggunakan algoritma cosinus dilakukan menggunakan program sederhana berbasis java. Dengan menggunakan fasilitas *library wordnet similarity for* java (WS4J), makan akan lebih mudah mengimplementasikan algoritma cosinus dengan bahasa java. Seluruh proses mulai dari *case folding* hingga perhitungan cosinus dilakukan dengan program berikut ini :



Gambar 2. Perhitungan Cosinus

Hasil perhitungan cosinus dari kalimat "Peneliti membuat program dengan bahasa java" dan "Peneliti melakukan *coding* dengan bahasa java" adalah 0,8 yang berarti kedua kalimat tersebut terdeteksi plagiat berdasarkan kemiripan kata menggunakan algorita cosinus.

Untuk menguji penggunaan algoritma wu palmer dalam studi kasus mendeteksi plagiasi, peneliti menggunakan situs ws4jdemo.appspot.com. hasil yang diperoleh adalah, wu palmer lebih mudah mendeteksi kata saja dalam *word net* nya bukan kalimat. Sehingga hasil yang diperoleh ketika menginputkan kalimat adalah tampilan "*invalid input*".

5. KESIMPULAN

Dari hasil penelitian dapat disimpulkan bahwa :

1. Cosinus bisa digunakan sebagai algoritma dalam mendeteksi plagiasi berdasarkan kemiripan kata.
2. Jika hasil nilai perhitungan cosinus 0 - 0,5 maka tidak plagiat, jika > 0,5 - 1 maka plagiat
3. Algoritma cosinus lebih efektif digunakan untuk mendeteksi plagiasi daripada algoritma wu palmer.

5. SARAN

Peneliti menyadari bahwa dalam penyusunan makalah ini banyak sekali keterbatasan,sehingga makalah ini jauh dari sempurna. Keterbatasan waktu dan yang lainnya menjadi penyebabnya, oleh karena itu peneliti menyarankan hal berikut untuk penelitian selanjutnya :

1. Pengembangan sistem *plagiarism checker* yang berbasis web dengan algoritma cosinus yang dikombinasikan dengan algoritma lainnya.
2. Penggunaan wu palmer untuk mencari *keyword* dalam *open jurnal system*.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Program Studi Sistem Informasi, Fakultas Sains dan Teknologi, UIN Sunan Ampel Surabaya selaku instansi tempat penulis berasal.

DAFTAR PUSTAKA

Akip Maulana, dkk. (2016). Perancangan *Semantic Similarity* Based on Word Thesaurus Menggunakan Pengukuran Omiotis untuk Pencarian Aplikasi

pada I-GRACIAS. Indonesia Symposium on Computing Telkom University, September 2016.

Anang, K. (2017). *Teknik Sitasi dalam Penulisan Karya Tulis.* Diakses dari http://www.uinsby.ac.id/uploads/2017/02.

Departemen Pendidikan Nasional. (2005). *Kamus Besar Bahasa Indonesia Edisi Ketiga*. Jakarta : Balai Pustaka.

Hanto dan Harianto Kristatanto. (2015). *Program Bantu Pemilihan Lagu Pujian Berdasarkan Tema Kebaktian dengan Menggunakan Metode Cosinus Similarity Studi Kasus: GKI Ngupasan.* Jurnal EKSIS, Vol 8, No 1, 2015.

Na'firul Hasna Ariyani,dkk. (2016). *Aplikasi Pendeteksi Kemiripan Isi Teks Dokumen Menggunakan Metode Levenshtein Distance*. Jurnal semanTIK, Vol 2, No.1, Januari – Juni 2016.

Ogie Nurdiana, dkk. (2016). *Perbandingan Metode Cosine Similarity dengan Metode Jaccard Similarity pada Aplikasi Pencarian Terjemah Al-Qur'an dalam Bahasa Indonesia.* Jurnal JOIN*,* Volume I, No. 1, Juni 2016.

Paratisa Kharismadita dan Faisal Rahutomo. (2015). *Implementasi Tokenizing Plus pada Sistem Pendeteksi Kemiripan Jurnal Skripsi*. Jurnal Informatika Polinema, Volume 2, Edisi 1, November 2015.

Radiant Victor Imbar, dkk. (2014). *Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks*. *Jurnal Informatika.* Jurnal JuTISI, Vol. 10 No. 1, Juni 2014, pg 31 - 42.