



PENGEMBANGAN APLIKASI PENDETEKSI PLAGIARISME PADA DOKUMEN TEKS MENGGUNAKAN ALGORITMA RABIN-KARP

Oleh

Kadek Versi Yana Yoga,0815051046
Jurusan Pendidikan Teknik Informatika
Fakultas Teknik dan Kejuruan
Universitas Pendidikan Ganesha
Email : ada_dogen@yahoo.co.id

ABSTRAK

Penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah aplikasi pendeteksi plagiarisme pada dokumen teks berbahasa Indonesia menggunakan algoritma rabin-karp. Inputan dari aplikasi ini berupa dokumen yang berformat *.txt, *.html, *.php, *.xml, *.doc, *.docx, *.rtf, dan keluaran dari aplikasi ini adalah berupa persentase kesamaan dari dua buah dokumen

Hasil dari penelitian ini disebut “PlagiaTor” yang dikembangkan dengan menggunakan bahasa pemrograman *Borland Delphi 7*. Aplikasi ini mampu mendeteksi plagiarisme pada dokumen berbahasa Indonesia pada kata yang bersinonim. Pada aplikasi “PlagiaTor” ini terdapat 5 proses untuk mendeteksi kesamaan dua buah dokumen yaitu : *casefolding dan tokenizing* , *filtering*, *synonym recognition*, *stemming dan string matching*. Pengujian dilakukan dengan mengujicobakan dokumen yang dipotong pada bagian tertentu serta mengganti kata dengan sinonimnya dan menggunakan *k-gram* yang berbeda. Berdasarkan pengujian tersebut, didapatkan hasil bahwa aplikasi sudah dapat mendeteksi plagiarisme pada kata yang bersinonim dengan baik

Kata kunci: *text mining*, Rabin-Karp, plagiarisme, *casefolding dan tokenizing* , *filtering*, *synonym recognition*, *stemming dan string matching*.



DEVELOPMENT OF PLAGIARISM DETECTION APPLICATION ON THE TEXT DOCUMENTS USING RABIN-KARP ALGORITHM

By

Kadek Versi Yana Yoga,0815051046
Jurusan Pendidikan Teknik Informatika
Fakultas Teknik dan Kejuruan
Universitas Pendidikan Ganesha
Email : ada_dogen@yahoo.co.id

ABSTRACT

This study was aimed to design and implement a plagiarism detection application on the Indonesian documents texts using Rabin-Karp algorithm. The input of this application such documents which have text file (*.txt, *.html, *.php, *.xml, *.doc, *.docx, *.rtf) and the output of this application is the percentage of similarity of two documents.

The result of this study is an application which is called "PlagiaTor" which was developed by using Borland Delphi 7 programming language. This application was able to detect plagiarism synonym words on Indonesian language document. On the PlagiaTor application there were five processes to detect the similarity of two documents, there were: casefolding and tokenizing, filtering, synonym recognition, stemming and string matching. The testing was done by trying out cutting document in any part of that document then the words were replaced by that synonym of those words and using different k-gram. Based on the testing, could be obtained the result that the application was able to detect plagiarism on synonym word well.

Key Word: text mining, string matching, Rabin-Karp, plagiarism, stemming, hashing, filtering, synonym recognition.

I. PENDAHULUAN

Dengan semakin berkembangnya teknologi informasi menyebabkan semakin mudahnya pertukaran informasi dewasa ini tidak hanya membawa dampak positif bagi kemajuan teknologi, tetapi juga membawa dampak negatif. Dampak negatif ini biasa dilakukan oleh mahasiswa, dimana mahasiswa dalam menyelesaikan tugas cukup dengan teknik *copy-paste* dan mengganti *cover* tugas milik teman. Meskipun telah berulang kali diberi peringatan, tetap saja praktik

penjiplakan dalam tugas bahkan ujian dilakukan oleh mahasiswa. Hal ini menimbulkan kemalasan pada mahasiswa dan juga masalah dalam tahap evaluasi terhadap hasil pembelajaran mahasiswa. Tidak mudah bagi seorang pengajar untuk memberikan penilaian objektif terhadap tugas atau ujian yang telah diberikan kepada mahasiswa. Akan semakin bertambah parah apabila mahasiswa yang mengikuti perkuliahan tersebut jumlahnya sangat banyak. Kemungkinan terjadinya penjiplakan akan semakin besar dan akan semakin sulit mendeteksinya mengingat begitu banyak dokumen teks yang terlibat.

Praktik penjiplakan ini termasuk kegiatan plagiarisme. Plagiarisme merupakan suatu tindakan menjiplak karya seseorang dan kemudian mengakuinya sebagai karya sendiri. Sebagai usaha untuk menekan terjadinya tindak plagiarisme sudah banyak dibuat sistem pendeteksi plagiarisme pada dokumen teks di antaranya yaitu: Pembuatan Program Aplikasi untuk Pendeteksian Kemiripan Dokumen Teks dengan Algoritma Smith-Waterman (Thalib,2010), namun aplikasi yang dibuat ini masih memiliki batasan masalah yaitu file input masih berupa *.txt dan belum bisa mendeteksi kemiripan pada kata yang bersinonim.

Berdasarkan uraian di atas, peneliti tertarik untuk mengembangkan sebuah pendeteksi plagiarisme pada dokumen teks yang dapat melengkapi kekurangan aplikasi sebelumnya yaitu dapat mendeteksi kemiripan pada kata yang bersinonim (*Synonym Recognition*) dan file input tidak hanya berupa file *.txt menggunakan algoritma Rabin-Karp yang berjudul “Aplikasi Pendeteksi Plagiarisme pada Dokumen Teks Menggunakan Algoritma Rabin-Karp”.

II. METODOLOGI

2.1. Pengertian Plagiarisme

Kata plagiarisme berasal dari kata Latin *plagiarius* yang berarti merampok, membajak. plagiarisme merupakan tindakan pencurian atau kebohongan intelektual. Plagiarisme didefinisikan dalam buku “Kode Etika Peneliti” (MPR LIPI, 2007) sebagai mengambil alih gagasan, atau kata-kata tertulis dari seseorang, tanpa pengakuan pengambilalihan dan dengan niat menjadikannya sebagai bagian dari karya keilmuan yang mengambil.

2.2. Text Mining

Text mining, yang juga disebut sebagai *Text Data Mining* (TDM) atau *Knowledge Discovery in Text* (KDT), secara umum mengacu pada proses ekstraksi informasi dari dokumen-dokumen teks tidak terstruktur (*unstructured*). Teks *mining* dapat didefinisikan sebagai penemuan informasi baru dan tidak diketahui sebelumnya oleh komputer, dengan secara otomatis mengekstrak informasi dari sumber-sumber teks tidak terstruktur yang berbeda. Kunci dari proses ini adalah menggabungkan informasi yang berhasil diekstraksi dari berbagai sumber (Tan,1999).

Ada empat tahap proses pokok dalam *text mining*, yaitu pemrosesan awal terhadap teks (*text preprocessing*), transformasi teks (*text transformation*), pemilihan fitur (*feature selection*), dan penemuan pola (*pattern discovery*).

Teks yang akan dilakukan proses teks *mining*, pada umumnya memiliki beberapa karakteristik diantaranya adalah memiliki dimensi yang tinggi, terdapat *noise* pada data, dan terdapat struktur teks yang tidak baik. Cara yang digunakan dalam mempelajari suatu data teks, adalah dengan terlebih dahulu menentukan fitur-fitur yang mewakili setiap kata untuk setiap fitur yang ada pada dokumen. Sebelum menentukan fitur-fitur yang mewakili, diperlukan tahap *preprocessing* yang dilakukan secara umum dalam teks *mining* pada dokumen, yaitu *case folding*, *tokenizing*, *filtering*, *stemming*.

2.2.1. Case folding dan Tokenizing

Case folding adalah mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf a sampai dengan z yang diterima. Karakter selain huruf dihilangkan dan dianggap *delimiter*. Tahap *tokenizing* / *parsing* adalah tahap pemotongan string input berdasarkan tiap kata yang menyusunnya.

2.2.2. Filtering

Filtering adalah tahap mengambil kata-kata penting dari hasil token. Bisa menggunakan algoritma *stoplist* (membuang kata yang kurang penting)

atau *wordlist* (menyimpan kata penting). *Stoplist / stopword* adalah kata-kata yang tidak deskriptif yang dapat dibuang dalam pendekatan *bag-of-words*.

2.2.1 Stemming (Algoritma Stemming Nazief & Adriani)

Stemming merupakan suatu proses mentransformasikan kata-kata yang terdapat dalam suatu dokumen ke kata-kata akarnya (*root word*) dengan menggunakan aturan-aturan tertentu. Algoritma yang dibuat oleh Bobby Nazief dan Mirna Adriani ini memiliki tahap-tahap sebagai berikut:

1. Cari kata yang akan distem dalam kamus. Jika ditemukan maka diasumsikan bahwa kata tersebut adalah *root word*. Maka algoritma berhenti.
2. *Inflection Suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. Jika berupa *particles* (“-lah”, “-kah”, “-tah” atau “-pun”) maka langkah ini diulangi lagi untuk menghapus *Possesive Pronouns* (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus *Derivation Suffixes* (“-i”, “-an” atau “-kan”). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a
 - a. Jika “-an” telah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “-k” juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 3b.
 - b. Akhiran yang dihapus (“-i”, “-an” atau “-kan”) dikembalikan, lanjut ke langkah 4.
4. Hapus *Derivation Prefix*. Jika pada langkah 3 ada sufiks yang dihapus maka pergi ke langkah 4a, jika tidak pergi ke langkah 4b.
 - a. Periksa tabel kombinasi awalan-akhiran yang tidak diijinkan. Jika ditemukan maka algoritma berhenti, jika tidak pergi ke langkah 4b.
 - b. For $i = 1$ to 3, tentukan tipe awalan kemudian hapus awalan. Jika *root word* belum juga ditemukan lakukan langkah 5, jika sudah maka algoritma berhenti. Catatan: jika awalan kedua sama dengan awalan pertama algoritma berhenti.
5. Melakukan *Recoding*.

6. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai *root word*. Proses selesai.

2.3. *Synonym Recognition*

Synonym Recognition atau pengenalan sinonim adalah pendeteksian plagiarisme melalui pendekatan sinonim. Dalam hal ini dokumen satu dibandingkan dengan dokumen lain dengan mendeteksi kata-kata yang mengandung sinonim sehingga tingkat kesamaan dapat dikatakan lebih akurat.

Pada proses pencocokan sinonim, kata dalam dokumen dicocokkan dengan kamus sinonim yang terdapat pada *database*. Jika kata tidak tersedia pada kamus sinonim, maka tidak akan melakukan proses *synonym recognition*.

2.4. Algoritma Rabin-Karp

2.4.1. *Fungsi Hashing*

Fungsi Hash adalah sebuah fungsi yang mengkonvert setiap string menjadi bilangan, yang disebut *hash value*. Algoritma Rabin-Karp didasarkan pada fakta jika dua buah string sama maka harga hash valuenya pasti sama..Gambar di bawah ini merupakan rumus untuk mencari hash value:

$$h = c_1 * b^{k-1} | c_2 * b^{k-2} | \dots | c_{k-1} * b | c_k$$

Gambar 2. 1 Rumus Mencari Hash Value.

Keterangan:

h: *hash value*

c: nilai ascii karakter

b: basis(bilangan prima)

k: banyaknya karakter

2.4.2. K-Grams

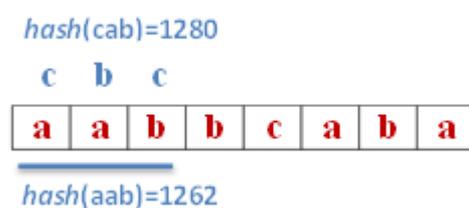
K-grams adalah rangkaian *terms* dengan panjang K. Kebanyakan yang digunakan sebagai *terms* adalah kata. *K-gram* merupakan sebuah metode yang diaplikasikan untuk pembangkitan kata atau karakter. Metode *k-grams* ini

digunakan untuk mengambil potongan-potongan karakter huruf sejumlah k dari sebuah kata yang secara kontinuitas dibaca dari teks sumber hingga akhir dari dokumen.

2.4.3. Prinsip Kerja Algoritma Rabin-Karp

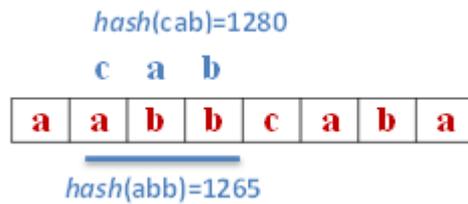
Pada dasarnya, algoritma Rabin-Karp akan membandingkan *hash value* dari string masukan dan substring pada teks. Apabila sama, maka akan dilakukan perbandingan sekali lagi terhadap karakter-karakternya. Apabila tidak sama, maka substring akan bergeser ke kanan.

Berdasarkan hasil penelitian yang dilakukan oleh Eko Nugroho dalam “*Perancangan Sistem Deteksi Plagiarisme Dokumen Teks Dengan Menggunakan Algoritma Rabin-Karp* (Nugroho, 2011)” didapat kesimpulan bahwa algoritma Rabin-Karp yang dimodifikasi (melakukan satu kali pengecekan yaitu pada *hash value*) mempunyai rata-rata waktu proses yang lebih baik dibandingkan dengan algoritma Rabin-Karp yang tidak dimodifikasi (melakukan dua kali pengecekan yaitu pada *hash value* dan karakter-karakternya), terutama dokumen teks yang mempunyai *size/ukuran file* yang besar, berdasarkan hal ini peneliti menggunakan algoritma Rabin-Karp modifikasi. Contohnya diberikan masukan “cab” dan teks “aabcbaba”. Langkah pertama cari *hash value* dari string “cab” berdasarkan rumus fungsi *hash* sehingga didapat: $hash(cab)=1280$.



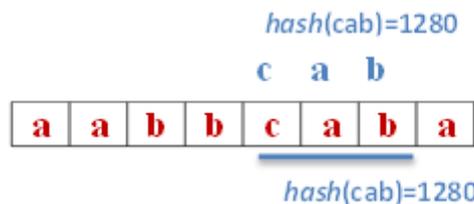
Gambar 2. 2 Fingerprint awal

Hasil perbandingan *hash value*nya ternyata tidak sama, maka substring pada teks akan bergeser satu karakter ke kanan.



Gambar 2. 3 Menggeser Fingerprint

Setelah mengalami pergeseran didapatkan *hash value* dari *fingerprint* “abb” =1265. Setelah itu dilakukan lagi proses perbandingan, dan mendapatkan hasil yang tidak sama. Begitu juga dengan perbandingan ketiga dan keempat tidak menghasilkan perbandingan yang sama. Pada perbandingan yang kelima didapatkan *hash value* yang sama.



Gambar 2. 4 Perbandingan kelima (string ditemukan)

Pada perbandingan yang kelima, kedua *hash value* dan karakter pembentuk string sesuai, sehingga solusi ditemukan.

2.4.4. Pengukuran nilai *similarity*

Perhitungan *similarity* dari kumpulan kata menggunakan *Dice’s Similarity Coefficient* untuk pasangan kata yang digunakan. Nilai similaritas tersebut dapat dihitung dengan menggunakan :

$$S = \frac{2C}{A + B}$$

Dimana S merupakan nilai *similarity*, dan C merupakan jumlah *k-grams* yang sama dari dua buah teks yang di bandingkan, sedangkan A, B merupakan jumlah *k-grams* dari masing masing teks yang dibandingkan.

III. PEMBAHASAN

3.1 Analisis Masalah dan Usulan Solusi

Dengan kemajuan teknologi informasi dan komunikasi sehingga membuat pembuatan karya tulis semakin mudah dan cepat. Hal tersebut dikarenakan

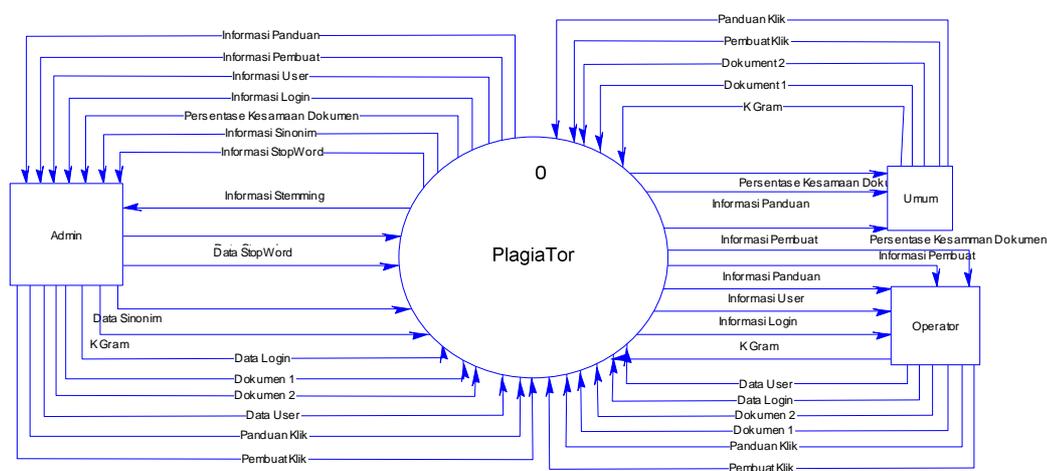
informasi kini tersedia secara melimpah. Akan tetapi dikarenakan kemudahan dalam memperoleh informasi tersebut, marak di temukan kesamaan karya tulis antara orang satu dengan yang lainnya, sehingga menimbulkan isu plagiarisme. Aksi plagiati dalam karya tulis sangatlah mungkin terjadi dan kasus plagiarisme paling banyak ditemukan di bidang akademisi, biasanya berupa esai, jurnal, laporan penelitian, dsb. Namun kini, plagiarisme dapat ditemukan hampir di semua bidang, oleh karena itu penting bagi peneliti untuk mengembangkan aplikasi dalam mendeteksi plagiarisme pada dokument teks.

3.2 Analisis Perangkat Lunak

Pada akhir dari penelitian skripsi ini, akan dibangun sebuah perangkat lunak berbasis *desktop*. Perangkat lunak yang dibangun mampu mendeteksi kesamaan dua buah teks berbahasa indonesia, diamana persentase kesamaan dua buah dokumen ini digunakan sebagai dasar dalam menentukan plagiarisme.

3.3 Perancangan Perangkat Lunak

Aplikasi yang akan dibangun menginputkan 2 buah document teks. Document yang dimasukkan kemudian dianalisis sehingga menghasilkan persentase kesamaan dari dua buah document tersebut, selain proses utama tersebut aplikasi ini juga menyediakan fasilitas *feedback* bagi pengguna untuk memperbaiki bila ada proses yang salah. Berikut adalah gambaran mengenai *Data Flow Diagram (DFD) Level 0*.

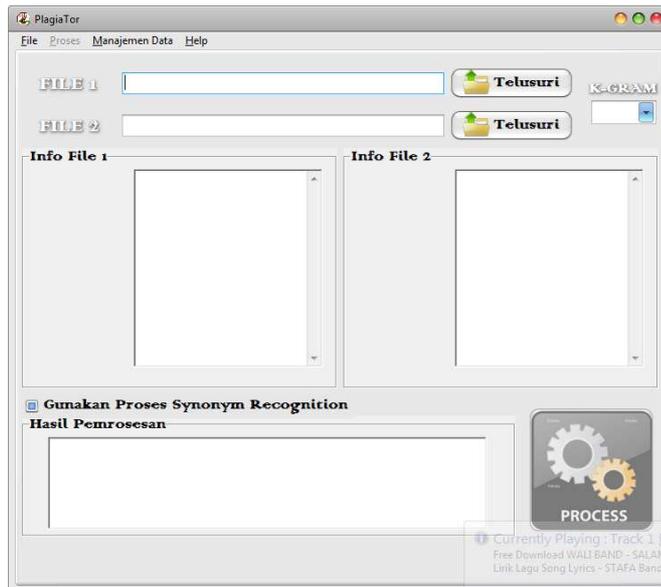


Gambar 3. 1 Diagram Konteks Aplikasi Pendeteksi Plagiarisme

IV. IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Perangkat Lunak *PlagiaTor*

Data Flow Diagram (DFD) PlagiaTor diimplementasikan dengan menggunakan bahasa pemrograman *Delphi 7*. Berikut ini tampilan *Form* Utama dari *PlagiaTor*.



Gambar 3. 2 Implementasi Form Utama Aplikasi Plagiator

4.2 Pengujian Perangkat Lunak *PlagiaTor*

Secara umum hasil pengujian fungsional menunjukkan bahwa sistem sudah dapat menangani data masukan yang tidak valid dan dapat menampilkan hasil persentase pendeteksian plagiarisme 2 buah document teks setiap proses dengan baik. Hasil pengujian konseptual menunjukkan bahwa sistem telah melaksanakan pendeteksian plagiarisme sesuai dengan apa yang diharapkan. Dibawah ini hasil dari pendeteksian plagiarisme:

Tabel 4. 1 Dokumen Uji

NO	Kode	Dokumen
1	A1	Laptop atau komputer jinjing adalah komputer bergerak yang berukuran relatif kecil dan ringan, beratnya berkisar dari 1-6 kg, tergantung ukuran, bahan, dan spesifikasi laptop tersebut. Sumber daya laptop berasal dari baterai atau adaptor A/C yang dapat digunakan untuk mengisi ulang baterai dan menyalakan laptop itu sendiri. Baterai

NO	Kode	Dokumen
		laptop pada umumnya dapat bertahan sekitar 1 hingga 6 jam sebelum akhirnya habis, tergantung dari cara pemakaian, spesifikasi, dan ukuran baterai. Laptop terkadang disebut juga dengan komputer notebook atau notebook saja. Sebagai komputer pribadi, laptop memiliki fungsi yang sama dengan komputer desktop (<i>desktop computers</i>) pada umumnya. Komponen yang terdapat di dalamnya sama persis dengan komponen pada desktop, hanya saja ukurannya diperkecil, dijadikan lebih ringan, lebih tidak panas, dan lebih hemat daya. Komputer jinjing kebanyakan menggunakan layar LCD (<i>Liquid Crystal Display</i>) berukuran 10 inci hingga 17 inci tergantung dari ukuran laptop itu sendiri. Selain itu, papan ketik yang terdapat pada laptop gratis juga kadang-kadang dilengkapi dengan papan sentuh yang berfungsi sebagai "pengganti" tetikus. Papan ketik dan tetikus tambahan dapat dipasang melalui soket Universal Serial Bus maupun PS/2 jika tersedia. Berbeda dengan komputer desktop, laptop memiliki komponen pendukung yang didesain secara khusus untuk mengakomodasi sifat komputer jinjing yang portabel. Sifat utama yang dimiliki oleh komponen penyusun laptop adalah ukuran yang kecil, hemat konsumsi energi, dan efisien. Komputer jinjing biasanya berharga lebih mahal, tergantung dari merek dan spesifikasi komponen penyusunnya, walaupun demikian harga komputer jinjing pun semakin mendekati desktop seiring dengan semakin tingginya tingkat permintaan konsumen
2	A2	Laptop atau komputer jinjing adalah komputer bergerak yang berukuran relatif kecil dan ringan, beratnya berkisar dari 1-6 kg, tergantung ukuran, bahan, dan detail laptop tersebut. Sumber daya laptop berasal dari baterai atau adaptor A/C yang dapat digunakan untuk mengisi ulang baterai dan menyalakan laptop itu sendiri. Baterai laptop pada umumnya dapat bertahan sekitar 1 hingga 6 jam sebelum akhirnya habis, tergantung dari cara pemakaian, detail, dan ukuran baterai. Laptop terkadang disebut juga dengan komputer notebook atau notebook saja. Sebagai komputer pribadi, laptop memiliki fungsi yang sama dengan komputer desktop (<i>desktop computers</i>) pada umumnya. Komponen yang terdapat di dalamnya sama persis dengan komponen pada desktop, hanya saja ukurannya diperkecil, dijadikan lebih ringan, lebih tidak panas, dan lebih irit daya. Komputer jinjing kebanyakan menggunakan layar LCD (<i>Liquid Crystal Display</i>) berukuran 10 inci hingga 17 inci tergantung dari ukuran laptop itu sendiri. kecuali itu, papan ketik yang terdapat pada laptop gratis juga kadang-kadang dilengkapi dengan papan sentuh yang berguna sebagai "alternatif" tetikus. Papan ketik dan tetikus tambahan dapat dipasang melalui soket Universal Serial Bus maupun PS/2 jika tersedia.
3	A3	Sebagai komputer pribadi, laptop memiliki fungsi yang sama dengan komputer desktop (<i>desktop computers</i>) pada umumnya. Komponen yang terdapat di dalamnya sama persis dengan komponen pada desktop, hanya saja ukurannya diperkecil, dijadikan lebih ringan, lebih tidak panas, dan lebih irit daya. Komputer jinjing kebanyakan menggunakan layar LCD (<i>Liquid Crystal Display</i>) berukuran 10 inci hingga 17 inci tergantung dari ukuran laptop itu sendiri. kecuali itu, papan ketik yang terdapat pada laptop gratis juga kadang-kadang dilengkapi dengan papan sentuh yang berguna sebagai "alternatif" tetikus. Papan ketik dan tetikus tambahan dapat dipasang melalui soket Universal Serial Bus maupun PS/2 jika tersedia. Berbeda dengan komputer desktop, laptop memiliki komponen pendukung yang didesain secara khusus untuk mengakomodasi sifat komputer jinjing yang portabel. Sifat utama yang dimiliki oleh

NO	Kode	Dokumen
		komponen penyusun laptop adalah ukuran yang kecil, irit konsumsi energi, dan efisien. Komputer jinjing biasanya berharga lebih mahal, tergantung dari label dan detail komponen penyusunnya, walaupun demikian harga komputer jinjing pun semakin mendekati desktop seiring dengan semakin tingginya tingkat permintaan pemakai.
4	A4	Laptop atau komputer jinjing adalah komputer bergerak yang berukuran relatif kecil dan ringan, beratnya berkisar dari 1-6 kg, tergantung ukuran, bahan, dan detail laptop tersebut. Sumber daya laptop berasal dari baterai atau adaptor A/C yang dapat digunakan untuk mengisi ulang baterai dan menyalakan laptop itu sendiri. Baterai laptop pada umumnya dapat bertahan sekitar 1 hingga 6 jam sebelum akhirnya habis, tergantung dari cara pemakaian, detail, dan ukuran baterai. Laptop terkadang disebut juga dengan komputer notebook atau notebook saja. Berbeda dengan komputer desktop, laptop memiliki komponen pendukung yang didesain secara khusus untuk mengakomodasi sifat komputer jinjing yang portabel. Sifat utama yang dimiliki oleh komponen penyusun laptop adalah ukuran yang kecil, irit konsumsi energi, dan efisien. Komputer jinjing biasanya berharga lebih mahal, tergantung dari label dan detail komponen penyusunnya, walaupun demikian harga komputer jinjing pun semakin mendekati desktop seiring dengan semakin tingginya tingkat permintaan pemakai.
5	A5	Laptop atau komputer jinjing adalah komputer bergerak yang berukuran relatif kecil dan ringan, , tergantung ukuran, bahan, dan detail laptop tersebut. Sumber daya laptop berasal dari baterai atau adaptor A/C yang dapat digunakan untuk mengisi ulang baterai dan menyalakan laptop itu sendiri. Baterai laptop pada umumnya dapat bertahan sekitar 1 hingga 6 jam sebelum akhirnya habis, tergantung dari cara pemakaian, detail, dan ukuran baterai. Laptop terkadang disebut juga dengan komputer notebook atau notebook saja. Sebagai komputer pribadi, laptop memiliki fungsi yang sama dengan komputer desktop (<i>desktop computers</i>) pada umumnya. Komponen yang terdapat di dalamnya sama persis dengan komponen pada desktop. Komputer jinjing kebanyakan menggunakan layar LCD (<i>Liquid Crystal Display</i>) berukuran 10 inci hingga 17 inci tergantung dari ukuran laptop itu sendiri. kecuali itu, papan ketik yang terdapat pada laptop gratis juga kadang-kadang dilengkapi dengan papan sentuh yang berguna sebagai "alternatif" tetikus Berbeda dengan komputer desktop, laptop memiliki komponen pendukung yang didesain secara khusus untuk mengakomodasi sifat komputer jinjing yang portabel. Komputer jinjing biasanya berharga lebih mahal, tergantung dari label dan detail komponen penyusunnya, walaupun demikian harga komputer jinjing pun semakin mendekati desktop seiring dengan semakin tingginya tingkat permintaan pemakai.
6	C	Menurut KBBI, pengertian komputer adalah alat elektronik otomatis yang dapat menghitung atau mengolah data secara cermat menurut yang diinstruksikan, dan memberikan hasil pengolahan, serta dapat menjalankan sistem multimedia, Hamacher, berpendapat bahwa komputer adalah mesin hitung elektronik yang cepat dengan menerima informasi input digital dan menghasilkan output berupa informasi. Saat ini, komputer sudah semakin canggih. Tetapi, sebelumnya komputer tidak sekecil, secanggih, sekeren dan seringan sekarang. Dalam sejarah komputer, ada 5 generasi dalam sejarah komputer. Generasi pertama dimulai saat perang dunia kedua yaitu

NO	Kode	Dokumen
		digunakan untuk mendesain pesawat terbang dan peluru kendali, generasi kedua dimulai tahun 1948 dengan digunakannya transistor, generasi ketiga dimulai dengan digunakannya chip pada computer, generasi keempat dimulai tahun 1980-an dengan digunakannya chip tunggal, dan generasi kelima sangat sulit karena tahap ini sangat muda.
7	D	Istilah komputer mempunyai arti yang luas dan berbeda bagi setiap orang, komputer berasal dari kata compute yang berarti alat untuk menghitung, menurut kamus oxford komputer artinya perangkat elektronik untuk menyimpan dan menganalisis informasi dimasukan ke dalamnya, untuk menghitung atau untuk mengendalikan mesin otomatis, Adapun komponen komputer adalah meliputi : Layar Monitor, CPU, Keyboard, Mouse dan Printer (sbg pelengkap). Tanpa printer komputer tetap dapat melakukan tugasnya sebagai pengolah data, namun sebatas terlihat dilayar monitor belum dalam bentuk print out (kertas). Dalam definisi seperti itu terdapat alat seperti slide rule, jenis kalkulator mekanik mulai dari abakus dan seterusnya, sampai semua komputer elektronik yang kontemporer. Istilah lebih baik yang cocok untuk arti luas seperti "komputer" adalah "yang memproses informasi" atau "sistem pengolah informasi."

Keterangan:

- Dokumen A1 merupakan dokumen asli
- Dokumen A2 : dokumen A1 dipotong bagian bawahnya dan diganti kata yang bersinonim dengan sinonimnya
- Dokumen A3 : dokumen A1 dipotong bagian atasnya dan diganti kata yang bersinonim dengan sinonimnya
- Dokumen A4 : dokumen A1 dipotong bagian tengahnya dan diganti kata yang bersinonim dengan sinonimnya
- Dokumen A5 : dokumen A1 dipotong pada bagian-bagian tertentu dan diganti kata yang bersinonim dengan sinonimnya.
- Dokumen C,D : dokumen dengan topic yang sama

Tabel 4. 2 Hasil Uji Aplikasi PlagaiTor dengan K-Gram 3

No	Kode Dokumen	Hasil	Menggunakan synonym recognition
1	A1 Dengan A1	100 %	100 %
2	A1 Dengan A2	77.9917469050894 %	81.0320781032078 %
3	A1 Dengan A3	80.2160702228224 %	83.1506849315068 %
4	A1 Dengan A4	72.2664735698769 %	75.2373995617239 %

5	A1 Dengan A5	84.5413726747915 %	88.8888888888889 %
6	C dengan D	59.2592592592593 %	57.8397212543554 %

Tabel 4. 3 Hasil Uji Aplikasi PlagiaTor Dengan K-Gram 4

No	Kode Dokumen	Hasil	Menggunakan synonym recognition
1	A1 Dengan A1	100 %	100 %
2	A1 Dengan A2	75.4820936639118 %	81.0055865921788 %
3	A1 Dengan A3	77.6200135226504 %	83.1275720164609 %
4	A1 Dengan A4	70.0507614213198 %	74.9085588880761 %
5	A1 Dengan A5	80.9248554913295 %	88.2238126219909 %
6	C dengan D	45.0116009280742 %	42.8405122235157 %

Tabel 4. 4 Hasil Uji Aplikasi PlagiaTor Dengan K-Gram 5

No	Kode Dokumen	Hasil	Menggunakan synonym recognition
1	A1 Dengan A1	100 %	100 %
2	A1 Dengan A2	72.9655172413793 %	80.979020979021 %
3	A1 Dengan A3	75.1523358158429 %	83.1043956043956 %
4	A1 Dengan A4	67.8286129266521 %	74.7252747252747 %
5	A1 Dengan A5	77.8135048231511 %	87.1661237785016 %
6	C dengan D	27.4418604651163 %	28.9381563593932 %

V. PENUTUP

Berdasarkan penelitian yang telah dilakukan yaitu “Pengembangan Aplikasi Pendeteksi Plagiarisme Pada Dokumen Teks Menggunakan Algoritma Rabin-Karp ” yaitu: 1) Rancangan aplikasi pendeteksi plagiarisme pada dokumen teks menggunakan algoritma rabin-karp menggunakan *Data Flow Diagram (DFD)* dipergunakan untuk menampilkan aliran data pada aplikasi *PlagiaTor*, 2) Implementasi Algoritma rabin-karp pada aplikasi aplikasi pendeteksi plagiarisme pada dokumen teks menghasilkan sebuah perangkat lunak yang mampu

menghasilkan aplikasi yang mendeteksi plagiarisme pada dokumen teks berbahasa Indonesia, 3) Dari hasil pengujian didapatkan hasil sebagai berikut: A) Aplikasi ini sudah dapat mendeteksi plagiarisme pada kata yang bersinonim B) Nilai K-Gram mempengaruhi hasil pendeteksian plagairme dimana semakin besar nilai K-Gram maka semakin diperhitungkan urutan kata dalam pendeteksian plagairime begitu juga sebaliknya.

VI. DAFTAR PUSTAKA

- Budi Cahyono, Eko et.al. 2010. *Aplikasi Pendeteksi Duplikasi Dokumen Teks Bahasa Indonesia Menggunakan Algoritma Winnowing Dengan Metode K-Gram Dan Synonym Recognition*. Jurusan Teknik Informatika Universitas Muhammadiyah Malang
- Firdaus, Hari Bagus. 2008. *Deteksi Plagiat Dokumen Menggunakan Algoritma Rabin-Karp*. Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung (ITB). Bandung.
- Kosinov, Serhiy. 2002. *Evaluation of N-Grams Conflation Approach in Text-Based Information Retrieval*. University of Alberta. Canada
- Nugroho, Eko. 2011. *Perancangan Sistem Deteksi Plagiarisme Dokumen Teks Dengan Menggunakan Algoritma Rabin-Karp*. Program Studi Ilmu Komputer, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Brawijaya.
- Thalib, Farid dan Kusumawati, Ratih. 2010. *Pembuatan Program Aplikasi untuk Pendeteksian Kemiripan Dokumen Teks dengan Algoritma Smith-Waterman*. Universitas Gunadarma.