

PENCARIAN KEMIRIPAN JUDUL SKRIPSI DAN ABSTRAK DENGAN METODE *EXACT MATCH* (STUDI KASUS PROGRAM STUDI TEKNIK INFORMATIKA UPN "VETERAN" YOGYAKARTA)

Heriyanto

Jurusan Teknik Informatika UPN "Veteran" Yogyakarta
Jl. Babarsari no 2 Tambakbayan 55281 Yogyakarta Telp (0274)-485323
e-mail : mr_heriyanto_skom@yahoo.com

Abstrak

Beberapa mahasiswa dalam pencarian tema dan judul skripsi sering kali mengalami kesulitan dan kendala dengan beberapa kali harus mengganti dan merubah judul skripsi. Pengajuan judul terkadang diterima dan ditolak berdasarkan pertimbangan salah satunya adalah judul yang sudah ada, Adapun kendala utama adalah menghindari indikasi adanya kesamaan judul dan adanya indikasi plagiat. Adanya kesamaan judul skripsi antara mahasiswa satu dengan lain tanpa dilakukan pengecekan dapat terjadi redundancy judul yang tentunya dapat dihindari apabila ada kontrol dari dosen atau jurusan yang dapat melakukan pengecekan sebelum judul tersebut di setujui. Berangkat dari akar adanya kesamaan dan kemiripan judul dan abstrak dan menjadikan indikasi plagiat maka perlu dibuatkan suatu sistem perangkat lunak yang secara otomatis dapat memudahkan melakukan pengecekan secara akurat pada judul-judul skripsi maupun abstrak mahasiswa.

Bertolak dari pentingnya adanya suatu perangkat lunak yang secara cepat dapat mengecek judul skripsi maupun abstrak maka penulis mengambil judul "Pencarian Kemiripan Judul Skripsi dan Abstrak dengan Metode Exact Match (Studi Kasus Program Studi Teknik Informatika UPN "Veteran" Yogyakarta) " dapat membandingkan sejauh mana nilai kemiripan dan beberapa judul yang sama dan serupa dapat tampil sebagaiantisipasi plagiat yang semakin marak.

Kata Kunci : kemiripan, skripsi, reducancy, akurat, pencarian, Exact Match.

1. PENDAHULUAN

Pencarian judul skripsi dalam penyelesaian tugas akhir mahasiswa merupakan sesuatu yang dapat dikatakan sulit dan dapat dikatakan mudah. Banyaknya judul yang sudah ada dengan adanya banyak kemiripan judul juga dapat diketahui dan dikurangi dengan melalui kontrol dan seleksinya dari para dosen, baik pembimbing dalam mengetahui secara dini apakah judul suatu skripsi tersebut sudah ada atau belum. Terkadang kemampuan dosen di dalam melakukan kontrol dan penyeleksian masih terkendala dengan suatu databases yang harus dicek dan diketahui dan dengan kemampuan daya ingat dari masing-masing dosen atau pembimbing, yang mungkin terbatas, sehingga terkadang satu dua judul terkadang lolos dan luput dari pantauan sehingga terjadi duplikasi judul dimana-mana.

Pencarian judul skripsi pada masing-masing institusi pendidikan harusnya dilakukan secara cermat dengan melakukan program aplikasi yang mudah untuk dioperasikan sehingga pada saat yang tepat dan cepat dapat diketahui bahwa judul tersebut sudah ada yang mirip sampai berapa persen dari masing-masing judul yang dilakukan pengujian.

Melalui pengolah data judul maka bagaimana suatu judul yang ada di dalam skripsi yang sudah terdokumentasi secara database dapat diketemukan pencarian judul yang mirip dapat diketahui secara cepat. masalah dengan membandingkan bagaimana mengidentifikasi judul skripsi dengan indeks suatu dokumen merupakan mirip antara judul skripsi satu dengan yang lain.

2. TINJAUAN PUSTAKA

Adapun beberapa penelitian yang sudah pernah dilakukan diantaranya :

1. Penerapan Algoritma Boyer-Moore untuk Pengecekan Plagiatisme Source Code (Redya Febriyanto) 2008

Penelitian pengecekan plagiatisme *source code* (Redya Febriyanto) Metode pencocokan *string* dapat diterapkan dalam berbagai macam hal. Salah satu contoh penerapannya adalah untuk membandingkan dua *source code* untuk mengetahui apakah terdapat kesamaan diantara kedua *source code* tersebut. Namun berbeda dari teks biasa, *source code* memiliki sifat-sifat khusus. Tingkat kesamaan antar *source code* tidak dilihat dari karakter per karakter pada teks tersebut, namun lebih ke kesamaan alur program.

2. TESSY (Test of Texts Similarity) 2008

Berdasarkan referensi dari aplikasi anticontek (Didi Achyari) 2008 mengembangkan TESSY. Namanya *Test of Texts Similarity*, atau disingkat TESSY. *Test of Texts Similarity* dibuat untuk mendeteksi kemiripan karya tulis mahasiswa. Fitur *Test of Texts Similarity* dikembangkan sebagai program desktop, bukan aplikasi berbasis web.

Karena setiap lembaga pendidikan umumnya punya standar kemiripan yang berbeda untuk dimasukkan dalam kategori penjiplakan, tingkat kemiripan yang ingin dideteksi lewat *Test of Texts Similarity* sengaja dibuat agar dapat diatur secara bebas. Aplikasi ini hanya mengecek kata-kata. Jadi, bisa membandingkan hasil karya tulis dengan dokumen yang berasal dari mana dan tahun kapanpun. Selain itu, *Test of Texts Similarity* juga dilengkapi dengan fitur untuk mencetak laporan formal.

3. **Penggunaan Metode *Exact Match* untuk Menentukan Kemiripan Naskah Dokumen Teks (Heriyanto 2011)**, Jurnal Telematika Volume 8, No. 1 Juli 2011. Melakukan perbandingan kemiripan naskah untuk beberapa dokumen teks yang analisa dilakukan berdasarkan kesamaan kata, letak dan padanan kata.
4. **Identifikasi Naskah Dokumen Teks Dengan Metode Indexing (Heriyanto, 2012)**, Jurnal Telematika Volume 8, No. 2 Januari 2012. Melakukan pengolahan data naskah teks dengan membuat indexing dengan melakukan scanning spasi, koma, titik, membuang spasi, mendeteksi suatu paragraph, pindah baris menghitung banyak kata yang muncul pada suatu kalimat dan menghitung jumlah data kata pada baris.

3. METODE PENELITIAN

Metode Penelitian

Penelitian ini disusun dengan beberapa tahapan mulai dari studi literatur sampai dengan pembuatan sistem untuk menguji hasil perbandingan dokumen sumber dan target.

Beberapa methods untuk mendeteksi plagiat secara umum (sumber *Plagiarism in natural and programming languages: an overview of current tools and technologies* (Clough P- July 2000) *Department of Computer Science, University of Sheffield*) yaitu :

1. Penggunaan kata
2. Perubahan kata
3. Korelasi teks
4. Frekwensi kata
5. Beberapa kata berbeda namun maksudnya sama seperti kata inggris makan ada banyak kata dalam manner.
6. Distribusi kata yang tersebar di setiap kalimat.
7. Syntak struktur dari teks

Namun yang dipakai yang menjadi acuan penulis hanyalah sebatas keberadaan kata-kata yang sama tepat dan letak kata serta padanan kata pada sumber dan target untuk melihat naskah itu mirip dengan sumber atau tidak.

Tahapan-tahapan tersebut sebagai berikut :

1. Studi Literatur
Penulis mencari sumber pustaka atau dokumen untuk mempelajari permasalahan seperti teori *Information Retrieval*, sehingga penulis memahami konsep IR, indeks dan ilmu pendukung lainnya. **Information Retrieval**. Sistem *Information Retrieval (IR)* dikembangkan sejak tahun 1940-an dengan tujuan untuk membantu pengolahan literatur ilmiah yang jumlahnya sangat banyak. Masalah pencarian informasi adalah mencari dokumen yang relevan dengan inputan pengguna seperti kata atau dokumen lain. Tujuan dari strategi pencarian otomatis (*automatic retrieval strategy*) adalah untuk memperoleh semua dokumen yang relevan dan pada saat yang sama mengambil sesedikit mungkin dokumen yang tidak relevan.
2. Tahapan Pengembangan Sistem
 - a. Tahap Perencanaan
Pada tahap kegiatan identifikasi masalah, mendefinisikan permasalahan, menyusun perencanaan.
 - b. Tahap Analisis
Analisa sistem adalah suatu proses untuk memahami sistem yang sudah ada dengan menganalisa permasalahan dan mencari solusi serta rencana-rencana pengolahan pencarian kata yang sama dan menganalisa seberapa mirip naskahnya.
 - c. Tahap Rancangan
Setelah mengetahui dan memahami masalah yang ada mencari solusi serta memahami kebutuhan pengguna dan mendesain sistem yang dapat berjalan untuk mencari kata yang sama dengan mengukur kemirip naskah.

- d. Tahap Impelentasi
Tahap ini dilakukan pembuatan algoritma pemrograman dan sistem yang telah dibuat menjadi program aplikasi. Pembuatan program aplikasi dapat mengetahui dokumen naskah sumber dan target mirip atau tidak.
- e. Tahap Evaluasi
Tahap pengujian terhadap sistem dilakukan dengan melakukan metode *exact match* dan menghitung perbandingan tingkat kemiripan dari dokumen naskah sumber dan target.

4. HASIL DAN PEMBAHASAN

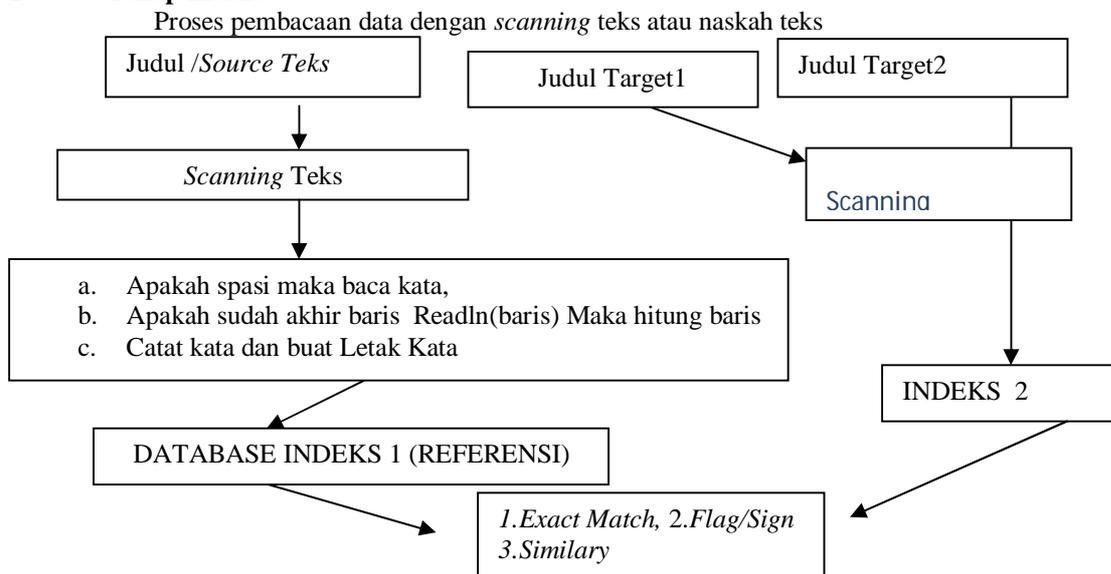
Sistem yang dibuat dan dirancang untuk melakukan pengujian judul skripsi baik naskah sumber teks 1 dan naskah sumber mteks 2 sebagai target dengan prediksi apabila ada kemiripan dari judul 1 dan judul 2 maka mendekati sama atau tidak sama. Secara garis besar sistem terdiri dari tiga proses yaitu proses awal (*praprocessing*), proses pencarian (*searching*) dan proses perhitungan presisi dengan *flag exact match*. Adapun kemampuan dan spesifikasi sistem sebagai berikut :

1. Dapat melakukan pencarian pada judul pertama dan judul lainnya, melakukan proses awal indeks 1 dan indeks 2 pada judul pertama dan judul kedua.
2. Sistem mampu menganalisa metode *exact match boolean (and)* dengan :
 - Kesamaan tanpa padanan kata maupun padanan kata dengan
 - Kesamaan kata dan letak berbeda
 - Kesamaan kata dan letak sama
3. Melakukan proses *flag* dari naskah pertama dan naskah kedua dengan *flag* 1 untuk kata yang tepat sama dan *flag* 0 untuk kata yang tidak sama.
4. Melakukan hasil *flag exact match* dari judul pertama dan judul berikutnya.
Membuat tampilan hasil perhitungan kemiripan seberapa besar perbedaannya ?

Melalui dokumen judul dapat melakukan indeks dengan menganalisa sebagai berikut :

- a. Data judul skripsi sumber sebagai referensi dapat dibuat indeks sekali saja yang dipakai untuk sebagai acuan pada data judul target
- b. Data judul target dapat dilakukan pengujian kepada sumber baik satu referensi maupun banyak referensi.
- c. Kemunculan membandingkan dengan hitung kata-kata sumber dan dengan kata-kata judul tujuan dapat diketahui seberapa mirip.

1. Tahap Indeks



Gambar 1 Proses Analisa Indeks dengan *Scanning*

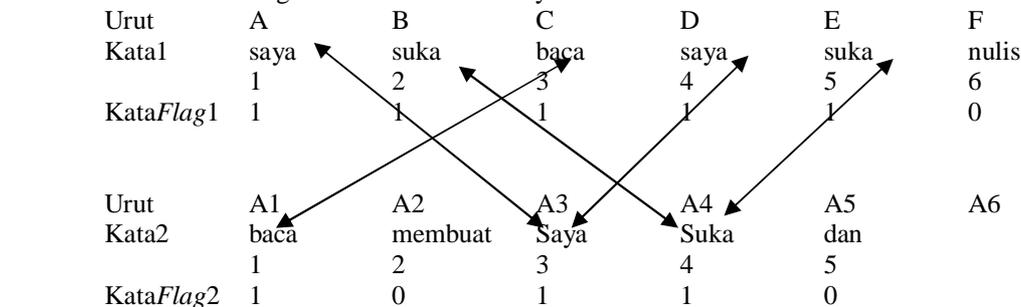
Data sumber merupakan data referensi yang dilakukan indeks dengan table indeks 1 terdiri dari *field* nom, nomkata, kodedok1, kata1, letak1. Nom untuk *auto number* merupakan *primary key*, nomkata merupakan nomor urutan kata, kodedok1 merupakan kode dokumen1 sebagai sumber dibuat kode, kata1 merupakan kata data sumber dan letak 1 merupakan letak dimana kata tersebut berada di dalam dokumen sumber. Pada indeks 2 merupakan indeks target yang terdiri dari nom, kodedok2, kata2, letak2. Nom untuk *Auto Number* pada indeks target, KodeDok2 digunakan kode dokumen target, kata2 dan letak2 untuk menyimpan indeks kata dan letak.

2. Tahapan Metode *Exact Match, Flag dan Similary*

Exact Match merupakan tahapan dilakukan untuk mengecek ketepatan kata jika ketemu sama maka kata tersebut tepat flag 1 dan jika tidak ketemu sama maka kata tersebut *Not Match*, flag 0, hitungan *similarity* menjumlahkan semua *flag* yang bernilai 1 (*true*) yang berasal dari *flag* 1. Hasil penjumlahan tersebut diambil nilai yang terkecil.

Contoh 1: Contoh kalimat sumber : Saya suka baca saya suka nulis

Contoh kalimat target : baca membuat saya suka dan



Tabel Sumber

Urut	A	B	C	D	E	F
KataFlag1	1	1	1	1	1	0

Tabel Target

Urut	A1	A2	A3	A4	A5	A6
KataFlag2	1	0	1	1	0	

Tabel Hasil

Sumber	1	1	1	1	1	0
Target	1	0	1	1	0	
Hasil	1	0	1	1	0	

Hasil = 1+0+1+1+0=3/6=0.5

Jika sama sumber=1 target=1 maka flag 1

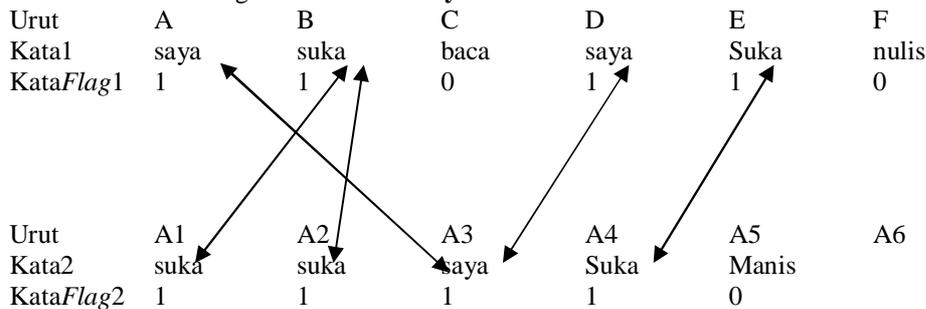
Jika tidak sama sumber=1 target=0 maka 0

Sumber=0 target=0 maka 0

Pada contoh 1 dapat dilihat bahwa pola kal 2 sangat hampir mirip kata dengan pola kasus sumber

Contoh 2: Contoh kalimat sumber : **Saya suka** baca **saya suka** nulis

Contoh kalimat target : suka suka **saya suka** manis



Tabel Sumber

Kata1	A	B	C	D	E	F
KataFlag1	1	1	0	1	1	0

Tabel Target

Kata2	A1	A2	A3	A4	A5	A6
KataFlag2	1	1	1	1	0	

Tabel Hasil

Sumber	1	1	0	1	1	0
Target	1	1	1	1	0	0
Hasil	1	1	0	1	0	0

Hasil = 1+1+0+1+0+0=3/6=0.5

Jika sama sumber bernilai =1 sedangkan target bernilai =1 maka *flag* adalah 1

Jika tidak sama walaupun sumber bernilai =1 sedangkan target=0 maka *flag* 0

Apabila sumber bernilai =0 sedangkan target=0 maka *flag* bernilai 0

Pada contoh 4 di atas dapat dilihat bahwa pola kalimat 2 sangat hampir mirip kata dengan pola kasus sumber

Contoh 5:

Contoh kalimat sumber : **Saya** suka baca **saya** suka gambar

Contoh kalimat target : **saya** makan nasi

Urut	A	B	C	D	E	F
Kata1	saya	suka	baca	Saya	Suka	Gambar
KataFlag1	1	0	0	1	0	0

Urut	A1	A2	A3	A4	A5	A6
Kata2	saya	makan	nasi			
KataFlag2	1	0	0			

Tabel Sumber

Kata1	A	B	C	D	E	F
KataFlag1	1	0	0	1	0	0

Tabel Target

Kata2	A1	A2	A3	A4	A5	A6
KataFlag2	1	0	0			

Tabel Hasil

Sumber	1	0	0	1	0	0
Target	1	0	0			
Hasil	1	0	0	0	0	0

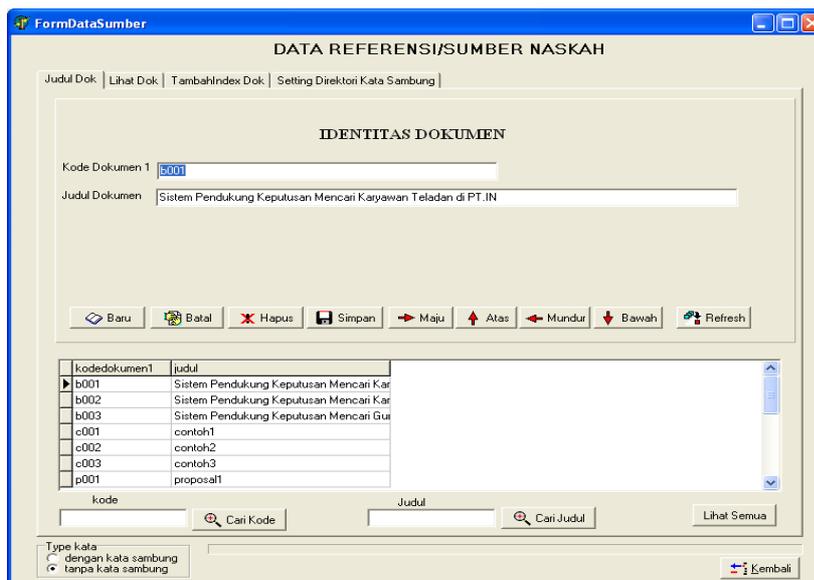
Hasil = $1+0+0+0+0+0=1/6=0.16$

Pada contoh 5 di atas dapat dilihat bahwa pola kalimat 2 sangat tidak mirip kata dengan pola kasus sumber. Semakin kecil tidak mendekati 1 maka pola kemiripan kasus akan semakin berkurang. Nilai 1 menyatakan sangat mirip (*exact match*) sedangkan nilai 0 menyatakan tidak mirip (*totally dissimilar*)

Pengambilan data sumber merupakan langkah awal untuk mengidentifikasi naskah sumber yang akan diambil. Data sumber dibuat sebagai referensi sehingga diperlukan data *master* yang terdiri dari *master* dokumen yang berisikan kode dokumen, judul dokumen, penulis, tahun terbit dan tanggal terbit.

3. Data Sumber

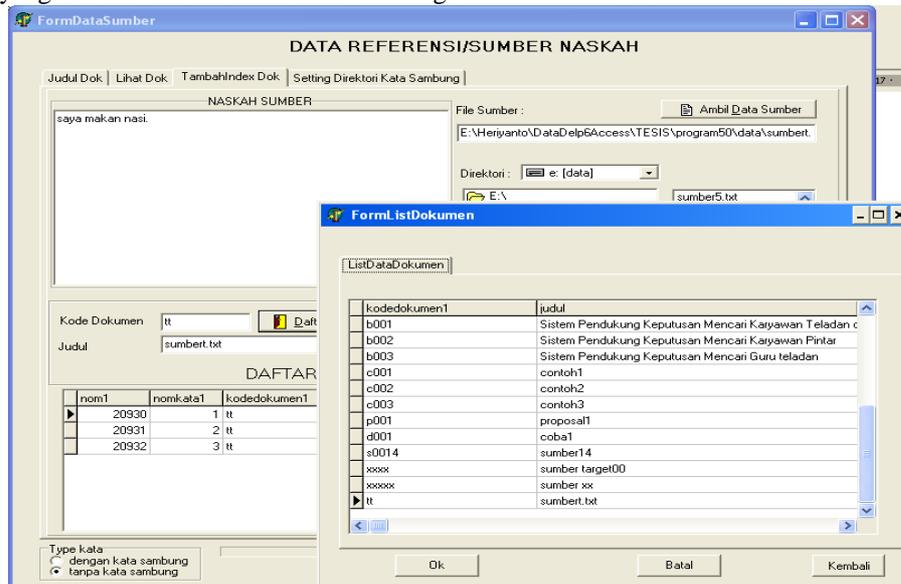
Membuat indek referensi pustaka dari suatu judul skripsi dalam bentuk indeks pada dokumen databases master.



Gambar 2. Data Sumber

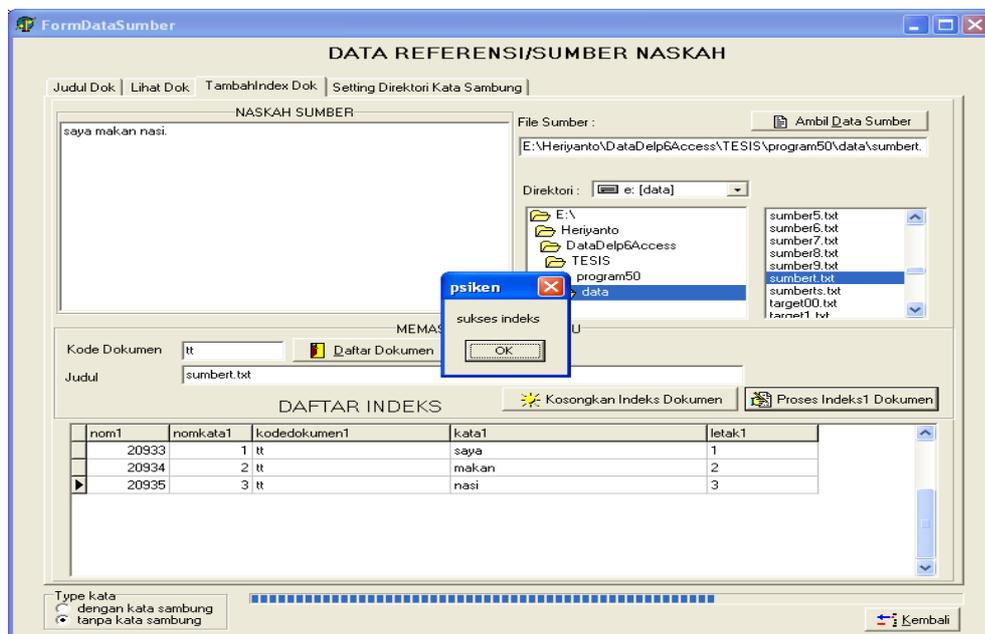
Dengan membuat referensi dalam bentuk *database* dokumen pada judul dokumen akan memudahkan di dalam penyimpanan sebagai acuan yang tentunya berupa kumpulan-kumpulan indeks dari berbagai sumber pustaka.

Sebelum melakukan indeks dilakukan pengambilan referensi dokumen dengan kodedokumen dalam daftar/ list sehingga apa yang akan diindeks akan direlasikan dengan *master* dokumen.



Gambar 3. Tambah Indeks Sumber Dokumen

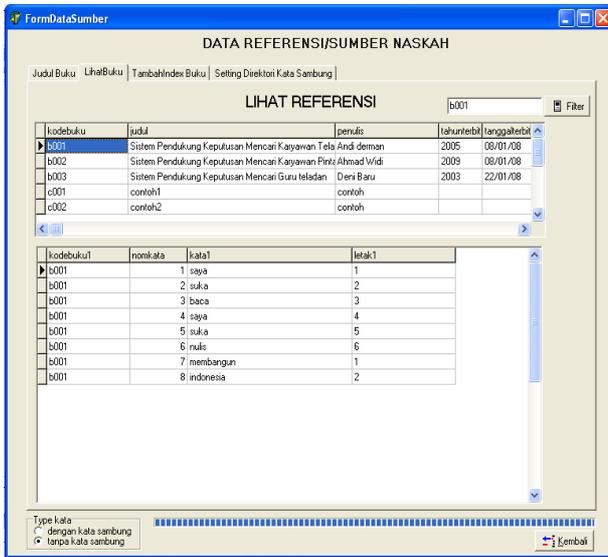
Memilih naskah sumber yang akan dijadikan indeks kemudian memilih *master* kode dokumen pada daftar dokumen dan selanjutnya adalah proses indeks. Begitu juga selanjutnya untuk sumber naskah yang lain juga dilakukan hal yang sama dengan memilih naskah sumber lain yang akan dijadikan referensi berupa *file* text lalu memilih *master* kode dokumen dan melakukan indeks. Pada tambah indeks akan menjadikan data sumber yang dapat diindeks hanya sekali dan ditambahkan berbagai naskah sumber yang lain. Hasil tampilan indeks 1 yang sudah selesai diproses dengan menghilangkan kata sambung sehingga akan memperkecil penyimpanan dalam daftar kata di indeks 1.



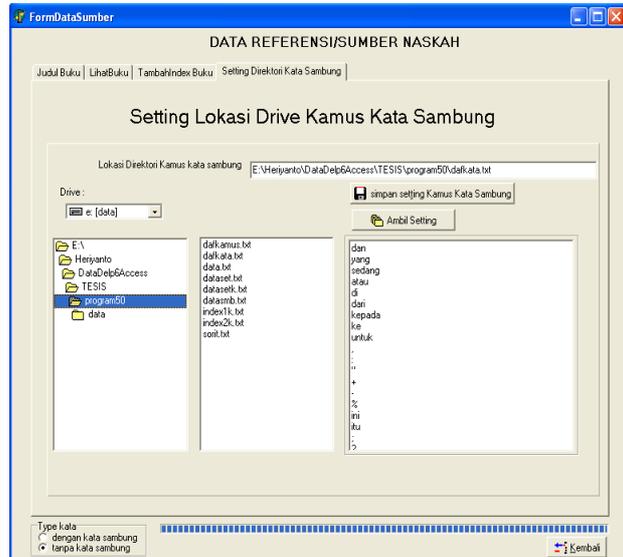
Gambar 4 Hasil Tampilan Indeks 1

Data yang sudah dilakukan indeks dapat dilihat dalam daftar indeks baik perdokumen maupun semua data yang sudah di indeks. Dalam indeks dilakukan penghilangan kata sambung yang sudah terdaftar dalam *menu* daftar kata sambung. Kata sambung tersebut dapat di tambahkan maupun dikurangi sehingga apabila menginginkan

kata sambung atau menginginkan simbol-simbol selain kata sambung ingin dihilangkan dapat dimasukkan dalam penyimpanan teks *dafkata.txt*



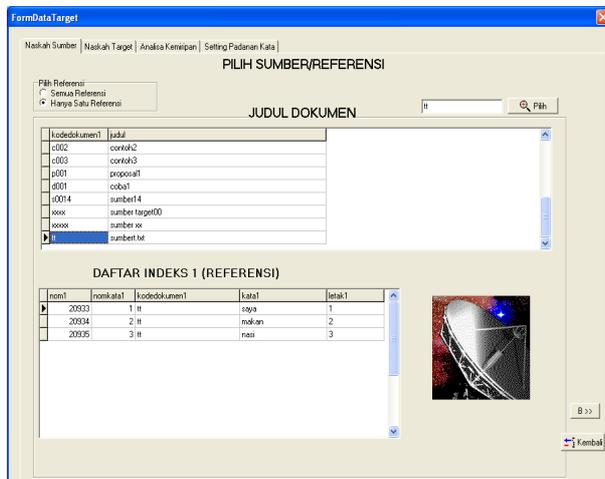
Gambar 5 Lihat Daftar Indeks Perdokumen



Gambar 6 Daftar Kata Sambung

4. Data Target

Naskah target merupakan naskah tujuan yang akan dibandingkan dengan naskah sumber. Pada prinsipnya sama dengan naskah sumber bahwa naskah target juga akan dilakukan indeks dengan nama indeks 2. Namun sebelum melakukan indeks pada naskah target terlebih dahulu dilakukan pemilihan pada naskah sumber/referensi apakah semua dokumen /referensi dipakai atau hanya satu referensi saja yang akan dipilih.

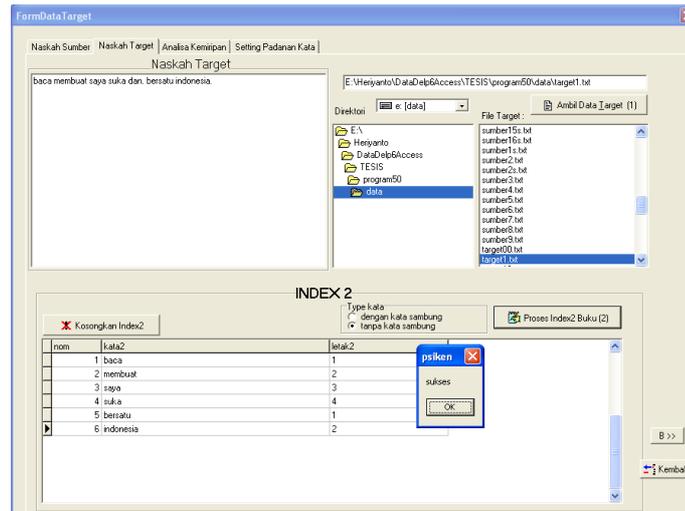


Gambar 7 Memilih Naskah Sumber



Gambar 8 Daftar Padanan Kata

Daftar padanan kata digunakan untuk analisa yang berdasarkan padanan kata. Daftar disimpan dalam *file* *dafkata.txt* yang didapat ditambah dan dikurang berdasarkan kesamaan padanan kata dapat lebih dari satu atau bahkan lebih padanan kata. Misalkan : ibu bapak orangtua
Selanjutnya adalah proses untuk melakukan indeks 2 pada naskah target dengan memilih *file* target berupa teks dan memproses sehingga menjadi indeks 2.



Gambar 9 Memilih Naskah Target dan Indeks Naskah Target

Pada analisa kemiripan digunakan untuk menganalisa dengan berdasarkan :

1. Tanpa padanan kata dengan kesamaan kata dan beda letak.
2. Tanpa padanan kata dengan kesamaan kata dan letak sama
3. Padanan kata dengan kesamaan kata dan beda letak
4. Padanan kata dengan kesamaan kata dan letak sama

Pada sumber dengan indeks 1 dan pada target dengan indeks 2. Pada sumber dengan padanan kata dengan indeks 1k dan pada target dengan padanan kata dengan indeks 2k

Tabel 1. Pengujian Judul Skripsi

Judul	A		B		C		D	
	Sumber	Target	Sumber	Target	Sumber	Target	Sumber	Target
Hasil	0,714	0,423	0,614	0,371	0,538	0,438	0,561	0,885

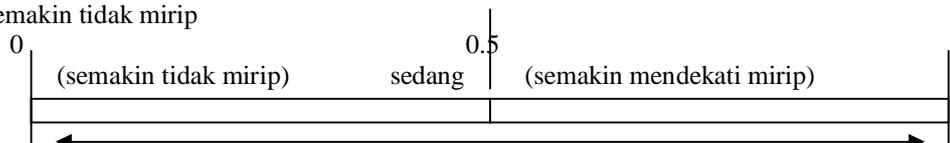
5. KESIMPULAN

Pembacaan dokumen judul 1 dan judul 2 sebagai teks sumber dan target dengan *type* teks dibaca kemudian dilakukan proses dengan memotong-motong perkata dan membuat indeks. Pada teks 1 diproses pemotongan kata menghasilkan indeks 1, begitu juga pada teks 2 juga diproses menghasilkan indek 2. Hasil indeks 1 dan hasil indeks 2 di simpan dalam *table*.

Proses berikutnya adalah dari hasil indeks 1 dan indeks 2 dibaca dan dilakukan metode pencarian match mencocokkan kata yang ada pada indeks 1 dengan kata yang ada pada indeks 2. Hasil proses pencocokan dari indeks 1 dan indeks 2 menghasilkan *flag* 1 dan *flag* 2. Jika terdapat kecocokan sama persis maka 1 jika tidak cocok maka 0. Hasil pengolahan yang terakhir adalah tabel *flag* 1 dan *flag* 2 dilakukan rekap sehingga proses representasi hitungan dilakukan pentotalan dan pembagian dengan hasil rata-rata ditampilkan pada seberapa nilai kesamaannya.

Semakin mendekati angka 1 maka naskah target dinyatakan semakin mirip tetap sebaliknya jika mendekati angka 0 maka semakin tidak mirip. Jika terletak ditengah-tengah 0.5 setengah mirip setengah tidak.

Dapat dilihat dalam bentuk garis semakin mendekati angka 1 semakin mirip sedangkan menjauh dari angka 0.5 semakin tidak mirip



Penggunaan program ini dapat dipakai pengujian suatu judul yang ada pada *file* teks sehingga apabila naskah tidak dalam bentuk format *script* teks dapat dilakukan import dengan beberapa *tool* atau melakukan *copy paste* dan disimpan dalam bentuk *file* text.

Kemiripan naskah sumber dan target berdasarkan analisa *exact match* tidak memperdulikan *stemming* sehingga apabila ada awalan dan akhiran tidak termasuk sama, sehingga kata dasar ditambah dengan awalan dan akhiran akan berarti kata yang berbeda-beda sehingga penulis menganggap tidak perlu dilakukan *stemming* karena pada akhir dan awalan memang kebanyakan kata selalu mempunyai arti berbeda.

DAFTAR PUSTAKA

- Achyari, Didi, 2008, *TESSY (Test of Texts Similarity)*
- Bouras, 2006, *The Importance Of The Difference In Text Types To Keyword Extraction: Evaluating A Mechanism*. URL http://ru6.cti.gr/ru6/publications/7699ICOMP2006_bouras.pdf
- Febriyanto, R, 2008, *Penerapan Algoritma Boyer-Moore untuk Pengecekan Plagiatisme Source Code*
- Heriyanto, 2011, *Penggunaan Metode Exact Match untuk Menentukan Kemiripan Naskah Dokumen Teks*, Jurnal Telematika Volume 8, No. 1 Juli 2011
- Heriyanto, 2012, *Identifikasi Naskah Dokumen Teks Dengan Metode Indexing*, Jurnal Telematika Volume 8, No. 2 Januari 2012
- Lu, G, 1999, *Multimedia Database Manajemen Systems*, Artech House, Inc
- Novia, W, *Kamus Lengkap Bahasa Indonesia*, Penerbit Kashiko Surabaya
- Reality, Tim, *Kamus Terbaru Bahasa Indonesia*, Penerbit Reality Publisher.
- Rhodes, B J., 2000., *Just-In-Time Information Retrieval, Ph.D. Thesis*, Massachusetts Institute of Technology, Cambridge, URL: <http://www.bradleyrhodes.com/papers/rhodes-phd-JITIR.pdf>
- Silberschatz, K, 2005, *Databases Systems Concept, 5th ed*, McGrawHill