

Pengembangan Sistem Penelusuran Katalog Perpustakaan Dengan Metode *Rocchio Relevance Feedback*

Pausta Yugianus, Harry Soekotjo Dachlan, dan Rini Nur Hasanah

Abstrak—Tanpa adanya sistem penelusuran katalog yang terkomputerisasi maka pengunjung perpustakaan mengalami kesulitan untuk mengetahui ketersediaan bahan pustaka dan pustakawan mengalami kesulitan dalam mengelola bahan pustaka. Tujuan penelitian ini yaitu mengimplementasikan metode *Rocchio relevance feedback* ke dalam sistem penelusuran katalog perpustakaan berbasis *web* dan menentukan proses penelusuran katalog perpustakaan berdasarkan input *term* dari *user*. Penelitian ini bermanfaat bagi pustakawan karena mempermudah pengelolaan data katalog dan bagi pengunjung karena mempermudah penelusuran katalog. Metode *rocchio relevance feedback* adalah metode modifikasi *query* yang memaksimalkan selisih antara rata-rata dokumen yang relevan dengan yang tak relevan. Metode *rocchio relevance feedback* terdiri dari empat tahap yaitu *tokenizing*, *filtering*, *stemming*, dan *term weighting*. Pada tahap *tokenizing*, *term* yang dimasukkan *user* dipisahkan menjadi kata tunggal. Pada tahap *filtering*, *term* penting akan diambil dan *term* yang tidak deskriptif akan dibuang. Pada tahap *stemming*, kata diubah menjadi akar kata dengan menghilangkan imbuhan. Pada tahap *term weighting*, *term* akan dihitung bobotnya untuk mengetahui ketersediaan *term* tersebut. Berdasarkan hasil penelitian disimpulkan bahwa sistem penelusuran katalog perpustakaan dengan mengimplementasikan metode *rocchio relevance feedback* dapat mempermudah pencarian pustaka dan menampilkan hasil yang memiliki nilai kemiripan tertinggi diantara dokumen yang ada sesuai dengan input *term* dari *user*.

Kata Kunci—penelusuran katalog perpustakaan, *rocchio relevance feedback*.

I. PENDAHULUAN

KATALOG perpustakaan merupakan alat bantu yang berfungsi untuk menunjukkan ketersediaan koleksi yang dimiliki oleh suatu perpustakaan. Sebagian besar perpustakaan masih menggunakan katalog fisik berupa kartu dan pencarian pustaka masih dilakukan secara manual tanpa bantuan komputer. Sistem manual tersebut menyebabkan kegiatan operasional perpustakaan berjalan lambat. Proses penelusuran katalog sangat penting untuk menghasilkan sebuah temuan atau informasi mengenai suatu bahan pustaka

Pausta Yugianus adalah Mahasiswa Program Magister Fakultas Teknik Universitas Brawijaya Malang (bwanna81@yahoo.co.id)

Harry Soekotjo Dachlan adalah Dosen Fakultas Teknik Universitas Brawijaya Malang (harrysd@ub.ac.id)

Rini Nur Hasanah adalah Dosen Fakultas Teknik Universitas Brawijaya Malang (rinihasanah@ub.ac.id)

yang relevan dan akurat. Dalam penelusuran informasi, jumlah dokumen relevan yang ditelusuri akan dipengaruhi oleh jumlah kata kunci yang digunakan [1].

Masalah yang sering terjadi dalam penelusuran katalog adalah pilihan kata yang digunakan pengguna dalam *query* seringkali berbeda dengan kata yang digunakan penulis dalam katalog perpustakaan, bahkan ketika beberapa *user* memiliki kebutuhan informasi yang sama, mereka jarang menggunakan pilihan kata yang sama untuk menggambarkannya dalam *query* [2].

Beberapa metode yang dapat digunakan dalam pencarian informasi yaitu *k-means clustering*, *widrow-hoff* dan *support vector machine* [3]. Penerapan *relevance feedback* untuk pencarian informasi dalam *database query* yang dievaluasi dan dioptimalkan oleh sistem *database* menunjukkan bahwa pencarian informasi menggunakan metode *relevance feedback* dapat mengurangi waktu pemrosesan 89.64% dari waktu pemrosesan dibandingkan tanpa menggunakan metode *relevance feedback* [3].

Penelitian mengenai pencarian informasi dengan menggunakan kata sebagai *query* awal dalam suatu sistem yang menggunakan metode *relevance feedback* dan tidak menggunakan *relevance feedback* menunjukkan bahwa pencarian informasi dari *query* yang dimasukkan *user* telah menghasilkan 73% dokumen yang akurat berdasarkan *query* dari *user* [4].

Penelitian ini bertujuan untuk implementasi metode *Rocchio relevance feedback* kedalam sistem penelusuran katalog perpustakaan berbasis *web* dan menentukan proses penelusuran katalog perpustakaan berdasarkan *input term* dari pengguna sistem.

Hasil yang diperoleh setelah melakukan pengujian dalam pengembangan sistem pencarian informasi interaktif dengan membandingkan metode *support vector machine* dan *rocchio relevance feedback* menunjukkan bahwa dengan jumlah data yang besar, metode *rocchio relevance feedback* lebih unggul dari metode *support vector machine*. Pencarian informasi menggunakan metode *rocchio relevance feedback* memiliki waktu pemrosesan lebih kecil 57.2% dan tingkat presisi 38.7% lebih besar dari *support vector machine* [5].

Pencarian informasi dengan membandingkan metode *k-means clustering* dan *rocchio relevance feedback* untuk pencarian berdasarkan *query* yang diberikan pengguna menunjukkan bahwa metode *rocchio*

relevance feedback lebih unggul dari metode *k-means clustering* dalam memberikan hasil pencarian kalimat yang berkaitan dengan *query* yang diberikan oleh *user*[6]. Metode *rocchio relevance feedback* memiliki akurasi 39.82% lebih besar dari metode *widrow-hoff*. Dari segi waktu penelusuran, metode *rocchio relevance feedback* memiliki waktu pencarian 67.41% lebih kecil dari metode *widrow-hoff*[7].

II. METODE ROCCHIO RELEVANCE FEEDBACK

Document retrieval adalah sebuah proses menggunakan komputer untuk menghasilkan daftar dokumen yang relevan dengan *request* dari *user* melalui perbandingan *request* dari *user* dengan indeks yang dihasilkan secara otomatis dari konten tekstual pada dokumen yang ada pada sistem[8]. Setiap dokumen dijelaskan oleh sekumpulan *keyword* yang merepresentasikan dokumen tersebut, yang disebut juga sebagai *index term*. *Index term* adalah sebuah kata (dokumen) yang secara semantik membantu untuk mengingat tema utama dari dokumen[9].

Metode *Rocchio relevance feedback* adalah strategi reformulasi *query* paling populer karena sering digunakan untuk membantu *user* pemula suatu *information retrieval systems*. Dalam siklus *relevance feedback*, kepada *user* disajikan hasil pencarian dokumen, setelah itu *user* dapat memeriksa dan menandai dokumen yang benar-benar relevan[10].

Teknik *relevance feedback* ditemukan pertama kali oleh Rocchio. Rocchio memandang *feedback* sebagai permasalahan mencari sebuah *query* optimal, yaitu *query* yang memaksimalkan selisih antara dokumen relevan dengan dokumen tak relevan. *Relevance feedback* berguna untuk mendekati *query* ke rataan dokumen relevan dan menjauhkan dari rataan dokumen tak relevan. Ini bisa dilakukan melalui penambahan istilah *query* dan penyesuaian bobot istilah *query* sehingga sesuai dengan kegunaan istilah tersebut dalam fungsinya membedakan dokumen relevan dan tak relevan [11].

Langkah-langkah dalam pengolahan *query* adalah sebagai berikut[12].

1) Tokenizing

Tokenizing adalah tahap pemrosesan sebuah kata kunci menjadi unit kecil yang disebut *token*. Unit tersebut dapat berupa suatu kata, suatu angka, atau suatu tanda baca.

2) Filtering

Tahap *filtering* adalah tahap mengambil kata-kata penting dari hasil *tokenizing*. Pada tahap ini kata-kata yang tidak deskriptif akan dibuang, misalnya “yang”, “dan”, “di”, “dari”.

3) Stemming

Tahap *stemming* adalah tahap mencari akar kata dari tiap kata hasil *filtering*. Pada tahap ini dilakukan proses pengembalian berbagai bentukan kata ke dalam suatu representasi yang sama, misalnya kata “menghasilkan” akan menjadi “hasil”.

4) Term Weighting

Pada tahap ini akan dilakukan penghitungan bobot tiap *term* yang dicari pada setiap dokumen sehingga dapat diketahui ketersediaan dan kemiripan suatu *term* di dalam dokumen.

Pada dokumen yang besar, skema yang paling sukses dan secara luas digunakan untuk pemberian bobot adalah skema pembobotan “*term frequency* * *inverse document frequency*” disebut sebagai nilai bobot *term* atau β . *Term frequency* (*tf*) adalah frekuensi dari kemunculan sebuah *term* dalam dokumen yang bersangkutan. *Idf* merupakan sebuah perhitungan dari bagaimana *term* didistribusikan secara luas pada koleksi dokumen yang bersangkutan. *Inverse document frequency* menunjukkan hubungan ketersediaan sebuah *term* dalam seluruh dokumen. Semakin sedikit jumlah dokumen yang mengandung *term* yang dimaksud, maka nilai *idf* semakin besar. Nilai *idf* sebuah *term* dirumuskan dalam persamaan berikut[13]:

$$idf = \log_{\frac{n}{df}} \quad (1)$$

Keterangan rumus (1) adalah sebagai berikut.

idf: nilai *inverse document frequency*;

n: jumlah dokumen di dalam koleksi;

df: nilai *document frequency*

Penghitungan bobot dari *term* tertentu dalam sebuah dokumen dengan menggunakan perkalian nilai *tf* dan *idf* menunjukkan bahwa deskripsi terbaik dari dokumen adalah *term* yang banyak muncul dalam dokumen tersebut dan sangat sedikit muncul pada dokumen yang lain. Perhitungan bobot *term* adalah sebagai berikut[13]:

$$\beta = (tf) * (idf) \quad (2)$$

Keterangan rumus (2) adalah sebagai berikut.

β : nilai bobot *term*;

tf: nilai *term frequency*

idf: nilai *inverse document frequency*.

Tingkat kemiripan *term* pada dokumen yang dicari dapat dituliskan pada persamaan *Rocchio relevance feedback* berikut ini[14]:

$$R = N + \beta \left(\frac{Dp}{Np} - \left(\frac{Dn}{Nn} \right) \right) \quad (3)$$

Keterangan rumus (3) adalah sebagai berikut.

R: Tingkat kemiripan *term*;

N: Jumlah *term* tiap dokumen;

β : nilai bobot *term*;

Dp: *Term* dari dokumen relevan;

Np: Jumlah dokumen relevan;

Dn: *Term* dari dokumen tak relevan;

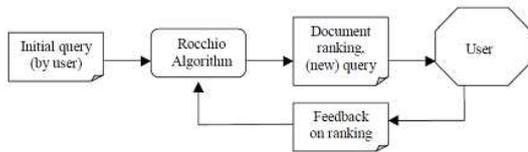
Nn: Jumlah dokumen tak relevan.

III. IMPLEMENTASI METODE ROCCHIO RELEVANCE FEEDBACK PADA PENELUSURAN KATALOG PERPUSTAKAAN

Implementasi metode *rocchio relevance feedback* terdiri dari beberapa tahap sebagai berikut.

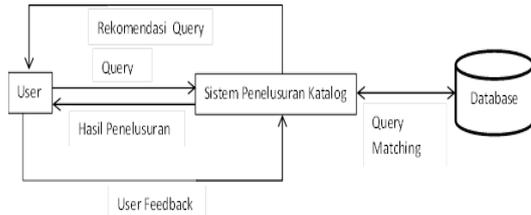
A. Konsep Penelusuran Katalog

Konsep penelusuran katalog perpustakaan yang akan dikembangkan adalah sebagai berikut.



Gambar 1. Konsep Rocchio Relevance Feedback

Gambar 1 menunjukkan bahwa metode *Rocchio relevance feedback* memberikan rekomendasi hasil penelusuran katalog kepada *user* dengan melakukan modifikasi *query* awal sehingga menghasilkan *query* yang lebih relevan sesuai dengan informasi yang ingin ditelusuri oleh pengguna.

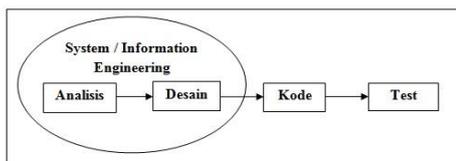


Gambar 2. Konsep Sistem Penelusuran Katalog Perpustakaan Dengan Metode Rocchio Relevance Feedback

Kinerja sistem penelusuran katalog yang akan dikembangkan adalah sebagai berikut. *User* memasukkan *query*, *query* yang dicari oleh *user* akan diolah oleh sistem untuk menghasilkan rekomendasi *query* kepada *user*, sehingga *user* dapat memilih *query* mana yang tepat seperti yang diinginkan *user*. Selanjutnya *user* memberikan *feedback* kepada sistem berdasarkan rekomendasi yang telah dipilih. Rekomendasi *query* yang dipilih oleh *user* akan ditelusuri oleh sistem.

B. Kerangka Solusi Penelitian

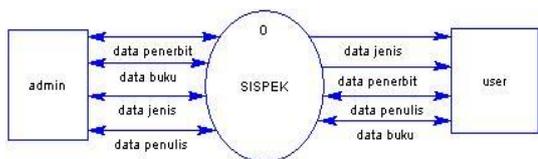
Pengembangan perangkat lunak sistem penelusuran katalog akan menggunakan model *waterfall*.



Gambar 3. Model Waterfall

Pengembangan aplikasi sistem penelusuran katalog menggunakan model sekuensial linier (*waterfall*) yang terdiri dari tahap analisis, desain, kode (implementasi) dan *test* (pengujian).

C. Pengembangan Perangkat Lunak

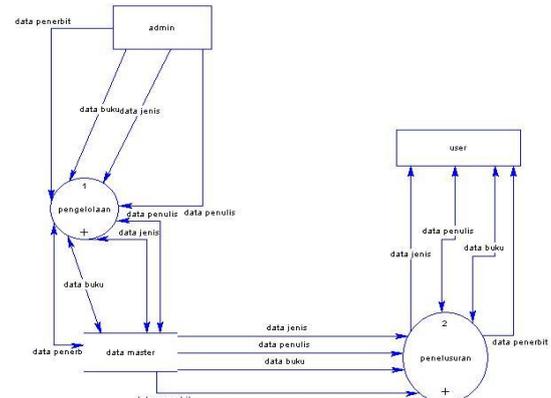


Gambar 4. Context Diagram Sistem Penelusuran Katalog

Pengembangan perangkat lunak sistem terdiri dari beberapa tahap yaitu tahap analisis, desain, implementasi, dan pengujian.

1) Analisis

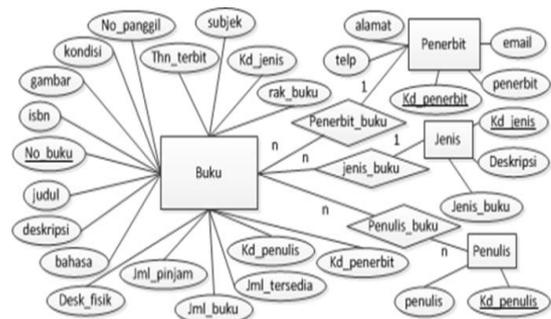
Admin dapat mengelola data katalog. *User* hanya dapat memasukkan kata kunci dan sistem akan memberikan hasil penelusuran.



Gambar 5. Data Flow Diagram Level 1 Sistem Penelusuran Katalog

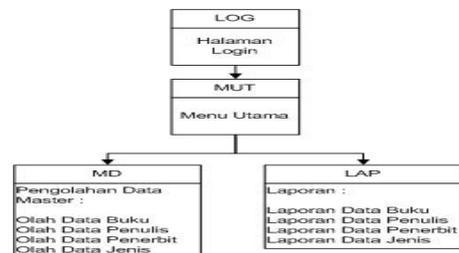
Dari diagram diatas terlihat bahwa terdapat dua proses utama yang menyusun sistem penelusuran katalog yaitu proses pengelolaan dan proses penelusuran.

2) Desain



Gambar 6. Rancangan Database Entity Relationship Diagram

Gambar 6 menggambarkan rancangan database sistem penelusuran katalog perpustakaan yang akan dikembangkan. Database sistem penelusuran katalog perpustakaan memiliki empat *entity* dan tiga *relationship*.



Gambar 7. Desain Antarmuka Admin

Struktur antarmuka admin terdiri dari halaman login, menu utama, pengelolaan data master dan laporan.



Gambar 8. Tampilan Antarmuka Login Admin

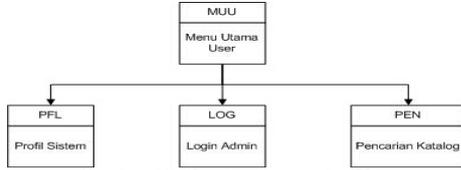
Admin dapat masuk ke halaman utama bila telah

melakukan verifikasi ID admin dan password.



Gambar 9. Tampilan Antarmuka Menu Utama

Pada halaman menu utama admin terdapat empat fitur yang dapat digunakan yaitu input data, pengelolaan data, laporan dan logout.



Gambar 10. Desain Antarmuka User

Struktur antarmuka user terdiri dari menu utama, profil sistem, login admin dan pencarian katalog.



Gambar 11. Tampilan Antarmuka Pencarian Katalog

Pada fitur pencarian katalog terdapat empat kategori pencarian data katalog, yaitu judul, pengarang, penerbit dan subjek.

3) Implementasi

Fitur untuk pengujung perpustakaan adalah fitur pencarian katalog. Katalog ditelusuri berdasarkan term dari pengguna. Term yang dimasukkan pengguna ke dalam sistem terdiri dari empat tipe yaitu judul, pengarang, penerbit, dan subjek. Hasil penelusuran katalog adalah sebagai berikut.

Judul	Subjek	Pengarang	Penerbit
Beban Tuntas Kisi-kisi Soal Ujian Nasional Biologi SMA	Ujian Nasional Biologi	Tin Harjanto	Visi Media
Jasa Biologi SMA Kelas 1, 2, 3	Biologi	Tajudin	Kawan Pustaka
Jasa Olimpiade Biologi SMA/MA	Olimpiade Biologi	Tomzi Zapino	Andi Publisher
Mandiri Biologi untuk SMA/MA Kelas X (Jilid 1)	Biologi	Sri Ayu Inaningtyas	Erlangga
Mandiri Biologi untuk SMA/MA Kelas XI (Jilid 2)	Biologi	Sri Ayu Inaningtyas	Erlangga
Mandiri Biologi untuk SMA/MA Kelas XII (Jilid 3)	Biologi	Sri Ayu Inaningtyas	Erlangga
Super Bank Soal Biologi SMA untuk Kelas 1, 2, dan 3	Bank Soal Biologi	Bersaki Umani	Wahyumanidin
Superlengkap Biologi SMA	Biologi	Siera Saenka Novel	Gagas Media

Gambar 12. Tampilan Hasil Penelusuran Katalog

Data katalog yang telah ditemukan dapat dilihat secara detail bila judul buku diklik seperti pada gambar dibawah ini.

membahas semua materi penting mengenai biologi untuk SMA secara ringkas dan sesuai kurikulum

Data Buku	Keterangan
Judul	Superlengkap Biologi SMA
No. Panggil	570 NOV s
Isbn	9797805972
Jenis	Ilmu Pengetahuan Alam
Subjek	Biologi
Pengarang	Siera Saenka Novel
Bahasa	Indonesia
Penerbit	Gagas Media
Tahun Terbit	2012
Deskripsi Fisik	viii, 316 p. il., 22,5 cm
Jumlah Buku	5
Jumlah Buku Yang Tersedia	5
Jumlah Buku Yang Dipinjam	0

Gambar 13. Tampilan Hasil Penelusuran Katalog Detail

Fitur yang disediakan untuk membantu pustakawan dalam mengelola data katalog perpustakaan adalah sebagai berikut.

- a) Input Data : Input buku, input deskripsi buku, input jenis buku, input penerbit, dan input penulis.
- b) Pengelolaan Data : Pengelolaan buku, pengelolaan deskripsi buku, pengelolaan jenis buku, pengelolaan penerbit, dan pengelolaan penulis.
- c) Laporan : Laporan buku, laporan deskripsi buku, laporan jenis buku, laporan penerbit, dan laporan penulis.



Gambar 14. Tampilan Fitur Input Buku

Pada fitur input buku, pustakawan dapat memasukkan data buku secara detail dan menyimpannya.



Gambar 15. Tampilan Fitur Pengelolaan Buku

Fitur pengelolaan buku memudahkan pustakawan untuk merubah atau menghapus data buku tertentu dengan menentukan judul buku yang ingin dihapus.



Daftar Buku										
Kode ISBN	Judul	No. Panggil	Subjek	Jenis	Penulis	Penerbit	Tahun Terbit	Jumlah Buku Tersedia	Uraian Buku Dipinjam	Deskripsi Fisik
00048786692921402	Pendidikan Agama Katolik Untuk SMA Kelas 1	588	Pendidikan Agama Katolik	Agama	Konrad KWI	Kanisius	2007	4	0	Indonesia, vi, 110 p., il., 21 cm
00048786692921403	Pendidikan Agama Katolik untuk SMA Kelas 2	589	Pendidikan Agama Katolik	Agama	Konrad KWI	Kanisius	2007	5	0	Indonesia, vi, 114 p., il., 21 cm
00048786692921405	Pendidikan Agama Katolik untuk SMA Kelas 3	590	Pendidikan Agama Katolik	Agama	Konrad KWI	Kanisius	2007	5	0	Indonesia, vi, 102 p., il., 21 cm

Gambar 16. Tampilan Fitur Laporan Buku

Fitur laporan buku memudahkan pustakawan untuk mengetahui koleksi buku secara detail.

4) Pengujian

Perangkat keras yang digunakan dalam proses pengujian adalah sebagai berikut.

- a. Processor Intel Centrino Duo 2.13 GHz
- b. RAM 4 GB
- c. Hard Disk 320 GB

Perangkat lunak yang digunakan dalam proses pengujian adalah sebagai berikut.

- a. Sistem Operasi Windows 7
- b. Browser Mozilla Firefox 19.0.2
- c. XAMPP 1.7.3.

Bentuk dari pengujian yang dilakukan adalah pengujian kebenaran fungsional unit perangkat lunak, pengujian dilakukan menggunakan teknik pengujian Black Box. Pengujian blackbox dilakukan untuk

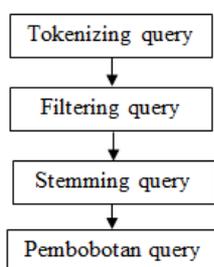
TABEL I.
HASIL PENGUJIAN PERANGKAT LUNAK

Deskripsi	Skenario	Hasil yang diharapkan	Hasil
Login Admin	1. Input Id admin tanpa input password admin	1. Tidak dapat masuk halaman administrator	OK
	2. Input Id admin benar dan password salah	2. Muncul peringatan password salah	
	3. Input id admin benar dan password benar	3. Masuk halaman administrator dan muncul konfirmasi id admin dan password valid	
Penelusuran katalog	1. Tidak menginput kata kunci katalog	1. Muncul peringatan untuk input kata kunci katalog	OK
	2. Input kata kunci katalog tanpa memilih option	2. Muncul peringatan untuk memilih option	
	3. Input kata kunci yang salah	3. Muncul peringatan bahwa data tidak ditemukan	
	4. Input kata kunci sesuai option	4. Tampil katalog sesuai input dan option	
Pengelolaan data master	1. Mengklik tombol input data tanpa memasukkan data	1. Muncul peringatan untuk memasukkan data terlebih dahulu, input data gagal	OK
	2. Input dan edit data master tetapi masih ada atribut yang tidak diisi	2. Muncul peringatan ada atribut yang belum diisi dan input atau edit data gagal	
	3. Input dan edit data master dengan tipe data yang salah	3. Muncul peringatan tipe data yang diinput tidak sesuai, input atau edit data gagal	
	4. Input dan edit data master dengan tipe data yang benar	4. Muncul peringatan input atau edit data berhasil	

menemukan kesalahan yang terjadi seperti fungsi yang tidak benar atau hilang, kesalahan antarmuka, kesalahan struktur data, kesalahan kinerja, atau kesalahan inisialisasi dan terminasi.

D. Pembahasan

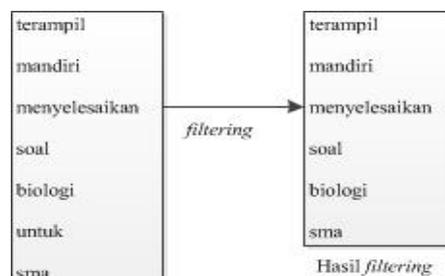
Proses pengolahan *query* pada sistem penelusuran katalog perpustakaan yang telah dikembangkan dapat digambarkan sebagai berikut.



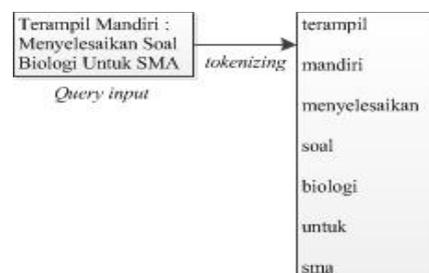
Gambar 17. Proses Pengolahan *Query*

Tahap *tokenizing* adalah tahap pemotongan string input berdasarkan tiap kata yang menyusunnya. Pada tahap ini dilakukan penghapusan tanda baca pada *query* yang dicari oleh *user* sehingga menghasilkan kata-kata yang berdiri sendiri.

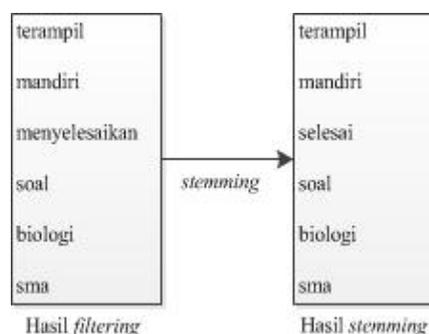
Tahap *filtering* adalah tahap mengambil kata-kata penting dari hasil *tokenizing*. Pada tahap ini sistem melakukan *stoplisting* (membuang kata yang kurang penting) dan *wordlisting* (menyimpan kata penting). *Stoplist/stopword* adalah kata-kata yang tidak deskriptif yang dapat dibuang. Contoh *stopwords* adalah “yang”, “dan”, “di”, “dari”, “ini”, “itu”, “untuk” dan sebagainya. *Filtering* dapat digambarkan sebagai berikut.



Gambar 19. Proses *Filtering*



Gambar 18. Proses *Tokenizing*



Gambar 20. Proses *Stemming*

Tahap *stemming* adalah proses mengubah kata menjadi kata dasarnya dengan menghilangkan imbuhan-imbuhan pada kata dalam dokumen atau mengubah kata kerja menjadi kata benda. *Stem* (akar kata) merupakan bagian dari kata yang tersisa setelah imbuhan (awalan dan akhiran) dihilangkan. Berikut ini adalah gambar proses *stemming*.

Tahap terakhir yaitu pembobotan *term*. Berikut adalah proses penghitungan bobot *term*.

Kata kunci (kk) : soal biologi sma

Dokumen 1: Cepat Menyelesaikan Soal Kimia SMA

Dokumen 2: Terampil Mandiri : Menyelesaikan Soal Biologi Untuk SMA

Dokumen 3: Juara Olimpiade Biologi SMA

Penghitungan modifikasi *query* berdasarkan persamaan *rocchio relevance feedback* pada Persamaan (3) adalah sebagai berikut.

Pada dokumen 1 :

$$R = 5 + 0.352 ((2 / 6) - (3 / 8)) = 4.985$$

Maka, tingkat kemiripan dokumen 1 terhadap kata kunci adalah 4.985.

Pada dokumen 2 :

$$R = 6 + 0.528 ((3 / 6) - (3 / 8)) = 6.066$$

Maka, tingkat kemiripan dokumen 2 terhadap kata kunci adalah 6.066.

Pada dokumen 3 :

$$R = 3 + 0.352 ((1 / 3) - (2 / 8)) = 3.029$$

Maka, tingkat kemiripan dokumen 3 terhadap kata kunci adalah 3.029.

Dari tabel diatas dapat disimpulkan bahwa Dokumen 2 (D2) memiliki tingkat kemiripan tertinggi diantara dokumen yang lain. Maka dari itu, dokumen 2 merupakan dokumen yang paling relevan berdasarkan kata kunci yang dicari oleh *user*.

TABEL II.
PEMBOBOTAN TERM

term	tf			df	n/df	idf	β (bobot)		
	D1	D2	D3				D1	D2	D3
cepat	1	0	0	1	3	0.477	0.477		
selesai	1	1	0	2	1.5	0.176	0.176	0.176	
soal	1	1	0	2	1.5	0.176	0.176	0.176	
Kimia	1	0	0	1	3	0.477	0.477		
sma	1	1	0	2	1.5	0.176	0.176	0.176	
terampil	0	1	0	1	3	0.477		0.477	
mandiri	0	1	0	1	3	0.477		0.477	
biologi	0	1	1	2	1.5	0.176		0.176	0.176
juara	0	0	1	1	3	0.477			0.176
olimpiade	0	0	1	1	3	0.477			0.176
Total bobot kata kunci pada tiap dokumen							0.352	0.528	0.352

TABEL III.
HASIL PERSAMAAN ROCCHIO RELEVANCE
FEEDBACK

	D1	D2	D3
R	4.985	6.066	3.029
Rank 2	Rank 1	Rank 3	

IV. KESIMPULAN

Dari hasil penelitian yang telah dilakukan maka diperoleh kesimpulan sebagai berikut.

- Proses penelusuran katalog didalam sistem penelusuran katalog perpustakaan menggunakan metode *rocchio relevance feedback* terdiri dari empat tahap yaitu tokenizing, filtering, stemming dan term weighting.
- Sistem penelusuran katalog perpustakaan metode *rocchio relevance feedback* dapat melakukan penelusuran katalog sesuai *input term* dari *user* yaitu judul, pengarang, penerbit dan subjek.
- Sistem penelusuran katalog perpustakaan dengan

mengimplementasikan metode *rocchio relevance feedback* menampilkan hasil penelusuran yang memiliki nilai kemiripan tertinggi diantara dokumen lain sesuai dengan *input term* dari *user*.

- Sistem penelusuran katalog perpustakaan memiliki fitur pengelolaan data katalog yang terdiri dari *input*, pengelolaan dan laporan data katalog perpustakaan sehingga mempermudah pengelolaan katalog.

DAFTAR PUSTAKA

- Spink, A., Jansen, B.J., Ozmutlu, H.C. "Use of Query Reformulation and Relevance Feedback by Excite Users". *International Journal on Internet Research*, Vol. 10 Iss: 4, pp.317-328. 2011.
- Xu, J., Croft, W.B. "Improving the Effectiveness of Information Retrieval with Local Context Analysis. *ACM Transactions on Information Systems*". 18(1):79-112. *Journal ACM Transactions on Information Systems (TOIS)*. New York, USA. Volume 18 Issue 1, Pages 79-112. 2011.
- Tari, Luis., Tu, Phan Huy., Hakenberg, Jorg., Chen, Yi., Son, Tran Cao., Gonzalez, Graciela., Baral, Chitta. "Incremental Information Extraction Using Relational Databases". *Journal IEEE Transactions on Knowledge and Data Engineering*, IEEE Educational Activities Department Piscataway, New Jersey, USA. Volume 24, Issue 1, pp. 86-99. 2012.
- Cooper, James W., Prager, John M. "Anti-Serendipity: Finding Useless Documents and Similar Documents". *Proceedings of the 33rd Hawaii International Conference on System Sciences*. Volume 3, Page 3008. 2012.
- Onoda, Takashi., Murata, H., Yamada, S. "Comparison of Performance for SVM based Relevance Feedback Document Retrieval in several Vector Space Models". *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, IEEE Computer Society, Washington DC, USA. Volume 03, Pages 169-172. 2011.
- Park, Sun., Cha, Byung Rae. "Query based Personalized Summarization Agent using NMF and Relevance Feedback". *Proceedings of the 4th International Conference on Uniquitous Information Management and Communication*, ICUIMC, New York, USA, Article No. 61. 2011.
- Shankar, Shrikanth., Karypis, George. "Weight adjustment schemes for a centroidbased classifier". *Proceedings of KDD Workshop on Text Mining*. Volume 6440 pp. 86-97. 2012.
- Liddy, E. D. *Automatic Document Retrieval*. In *Encyclopedia of Language and Linguistics*. 2nd Edition. Philadelphia:Elsevier Press. 2011.
- Yates, R.B., Neto, B.R. *Modern Information System*. Harlow : Addison Wesley. 2010.
- Selberg, E.W. *Information Retrieval Advances Using Relevance Feedback*. Thesis. Department of Computer Science And Engineering University of Washington. 2011.
- Ruthven, Ian., Lalmas, Mounia. "A Survey on the Use of Relevance Feedback for Information Access Systems". *The Knowledge Engineering Review*. Volume 18, Issue 02, pp. 95-145. 2011.
- Mooney, Raymond J. *Machine Learning Text Categorization*. Thesis. Department of Computer Sciences University of Texas. 2011.
- Uden, Mark Van. *Rocchio Relevance Feedback in Learning Classification Algorithms*. Thesis. Department of Computing Science University of Nijmegen. 2011.
- Potnis, Poorva. *Relevance Feedback in Flexible Retrieval Environment*. Thesis. The Faculty of Graduate School Department of Computer Science of University of Minnesota. 2011.