

APLIKASI PENCARIAN BARANG DENGAN ALGORITMA *RELEVANCE FEEDBACK* DI TOKO MEGA KOMPUTER BENGKULU

Siswanto¹, Indra Kanedi²

1,2 Jurusan Sistem Informasi, Fakultas Ilmu Komputer, Universitas Dehasen Bengkulu
Jl. Meranti Raya No.32 Sawah Lebar Kota Bengkulu, 38228

Email: sisunived@gmail.com, indrakanedi12@gmail.com

ABSTRAK

Aplikasi pencarian barang dengan algoritma *relevance feedback* di Toko Mega Komputer Bengkulu. Dimana Toko Mega Komputer menjual alat-alat komputer dan menyediakan jasa service komputer. Sistem penjualan di Toko Mega Komputer belum memanfaatkan aplikasi setiap penjualan yang terjadi setiap harinya, terkadang pihak Toko sering sekali kesulitan dalam mencari barang serta stok barang yang ada di Toko Mega Komputer.

Membuat aplikasi pencarian barang pada Toko Mega Komputer Bengkulu merupakan suatu aplikasi yang digunakan untuk membantu pihak toko dalam mengelola barang yakni data barang dan penjualan barang dengan metode *rocchio relevance feedback* menggunakan bahasa pemrograman *Visual Basic.Net* dan *Database SQL Server*.

Pada aplikasi pencarian data barang batas kata yang diperbolehkan hanya 4 (empat) kata, dan pencarian data barang berdasarkan per token. Proses pencarian data barang akan melalui 4 (empat) tahapan yaitu *Tokenizing*, *Stemming*, *Filtering*, *Weighting*. Berdasarkan hasil pengujian sistem, dapat disimpulkan aplikasi pencarian barang dapat membantu proses pengolahan data barang yang ada di Toko Mega Komputer.

Kata Kunci : Pencarian Barang, Algoritma Relevance Feedback, *Visual Basic.Net*

ABSTRACT

Goods search application with relevance feedback algorithm in the Mega Computer Store in Bengkulu. Where Mega Computer Stores sell computer equipment and provide computer services. The sales system at the Mega Komputer Shop has not utilized the application of every sale that occurs every day, sometimes the Stores often have difficulty in finding goods and stock of goods in the Mega Computers Store.

Make a search application for goods in the Bengkulu Mega Computer Store is an application that is used to assist the store in managing goods namely goods data and sales of goods with the method of rocchio relevance feedback using the Visual Basic.Net programming language and SQL Server Database.

In the item data search application the word limit is only 4 (four) words allowed, and item data search is based on per token. The process of finding data items will go through 4 (four) stages, namely Tokenizing, Stemming, Filtering, Weighting. Based on the results of system testing, it can be concluded that the application of searching goods can help the data processing of goods in the Mega Computer Store.

Keywords: Product Search, Relevance Feedback Algorithm, Visual Basic.Net

Pendahuluan

Aplikasi saat ini merupakan perangkat yang tidak bisa dipisahkan dari pekerjaan baik itu dalam skala kecil, sedang ataupun besar. Informasi diharapkan dapat mempercepat, otomatis serta tepat tujuannya dapat tercapai secara optimal dan maksimal. Hal demikian merupakan sudah menjadi kebutuhan yang positif bagi perkembangan usaha sendiri maupun kelompok itu sendiri, dimana masing-masing usah ingin maju lebih cepat dan lebih baik dari yang lain. Aplikasi atau disebut software mulai menjamur pada era informasi ini tentunya yang di kelola dan disampaikan terus di kembangkan dan informasi yang sederhana seperti sekedar menggambarkan suatu keadaan. Dengan dilihatnya perkembangan teknologi informasi tersebut, kita diajarkan dalam sejarah teknologi informasi dalam upaya untuk mendapatkan keutuhan ilmu dan pengetahuan tentang teknologi informasi.

Toko Mega Komputer merupakan salah satu toko komputer yang menjual alat-pemrograman VB.Net. Bertujuan untuk membuat Aplikasi Pencarian Barang Menggunakan Algoritma *Relevance Feedback* di Toko Mega Komputer agar Dapat membantu mengelola data barang, penjualan barang, nota penjualan, serta pencarian barang di Toko Mega Komputer

Tinjauan Pustaka

Barang adalah setiap benda baik berwujud maupun tidak berwujud, baik bergerak maupun tidak bergerak, dapat dihabiskan maupun tidak dapat dihabiskan, yang dapat untuk diperdagangkan, dipakai, dipergunakan, atau dimanfaatkan oleh konsumen.

Algoritma *Relevance Feedb* Teknik *Relevance Feedback* ditemukan pertama kali oleh *Rocchio* dan diterapkan dalam *vector space model*. *Rocchio* memandang algoritma kembali sebagai permasalahan mencari sebuah *query* optimal, yaitu *query* yang memaksimalkan selisih antara vektor rata-rata

alat komputer dan menyediakan jasa service komputer. Selama ini sistem penjualan di Toko Mega Komputer belum memanfaatkan aplikasi sebagai media pengolahan data barang, penjualan yang terjadi setiap harinya. Oleh karena itu, terkadang pihak Toko sering sekali kesulitan dalam mencari barang serta stok barang yang ada di Toko Mega Komputer. Fitur pencarian barang yang memudahkan pihak toko untuk mengetahui ketersediaan stok barang di Toko Mega Komputer.

Salah satu metode yang dapat digunakan untuk melakukan pencarian data adalah Algoritma *Relevance Feedback*, dimana hasil pencarian data didapat dari tahapan algoritma tersebut. Ada puncangkupan dalam pembuatan aplikasi ini yaitu Algoritma *Relevance Feedback* yang digunakan adalah *Rocchio*, *Relevance Feedback*, Fitur dari aplikasi ini yaitu pengolahan data barang, penjualan barang, nota penjualan, serta pencarian barang, Aplikasi ini dibuat menggunakan bahasa

dokumen relevan dengan vektor rata-rata dokumen tak-relevan. *Relevance Feedback* berguna untuk mendekatkan vektor *query* ke vektor rata-rata dokumen relevan dan menjauhkan dari vektor rata-rata dokumen tak relevan

Sistem *feedback* informasi terutama berhubungan dengan pencarian informasi yang isinya tidak memiliki struktur. Demikian pula ekspresi. Kebutuhan pengguna yang disebut *query*, juga tidak memiliki struktur. Hal ini yang membedakan sistem temu kembali informasi dengan sistem basis data. Dokumen adalah contoh informasi yang tidak terstruktur. Isi dari suatu dokumen sangat tergantung pada pembuat dokumen tersebut. *Relevance feedback* merupakan proses formulasi ulang *query* awal berdasarkan informasi umpan balik relevansi dari pengguna terhadap dokumen hasil pencarian awal.

Bagian-bagian dari sistem temu kembali informasi meliputi :

1. *Text Operations* (operasi terhadap teks) yang meliputi pemilihan kata-kata dalam *query* maupun dokumen (*term selection*) dalam pentransformasian dokumen atau *query* menjadi *terms index* (indeks dari kata-kata).
2. *Query formulation* (formulasi terhadap *query*) yaitu memberi bobot pada indeks kata-kata *query*.
3. *Ranking* (perangkingan), mencari dokumen-dokumen yang relevan terhadap *query* dan mengurutkan dokumen tersebut berdasarkan kesesuaiannya dengan *query*.
4. *Indexing* (pengindeksan), membangun basis data indeks dari koleksi dokumen. Dilakukan terlebih dahulu sebelum pencarian dokumen dilakukan.

Algoritma 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 isi tekstual pada dokumen yang ada pada sistem. Setiap dokumen dijelaskan oleh sekumpulan *keyword* yang merepresentasikan dokumen tersebut, yang disebut juga sebagai *index term*. *Index term* adalah sebuah kata yang secara semantik membantu untuk mengingat tema utama dari dokumen.

Metode Rocchio relevance feedback

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. 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 mendekatkan *query* ke dokumen yang relevan dan menjauhkan dari dokumen tidak 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. Langkah-langkah dalam pengolahan *query* :

- a. *Tokenizing* adalah tahap pemrosesan sebuah kata kunci menjadi unit kecil yang disebut *token*.
- b. 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”.
- c. 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”.
- d. *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* * *inversi 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

$idf = n \cdot df$

Keterangan 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 :

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

Keterangan 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] : $R = \beta \cdot (tf + idf)$

Keterangan rumus (3) adalah sebagai berikut.

R : Tingkat kemiripan term ;

N : Jumlah term tiap dokumen ;

β : nilai bobot term ;

Metodologi

Metode yang diterapkan membangun aplikasi ini metode **Metode Rocchio relevance feedback** 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. 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 mendekatkan *query* ke dokumen yang relevan dan menjauhkan dari dokumen tidak 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.

Penerapan Algoritma Relevance Feedback (Rocchio).

Algoritma *relevance feedback* yang digunakan dalam penelitian ini adalah *Rocchio Relevance Feedback*, dimana terdiri dari 4 tahapan yaitu *Tokenizing*, *Filtering*, *Stemming*, dan *Term Weighting*.

Tabel 1. Data Barang

Kod	Nama Barang	Sto
B001	Mouse Logitech	5
B002	Mouse Rexus	6
B003	Combo Mouse	4
B004	Keyboard Logitech	3
B005	Pointer Targus	7
B006	Joystik Logitech	4

Dari data tersebut, akan dicari barang menggunakan 2 kata kunci yaitu *Mouse Logitech*. Adapun tahapan-tahapan penyelesaian, antara lain :

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. Pada tahap ini, kata kunci akan diubah ke dalam bentuk *token* : *Token 1 = Mouse*

Token 2 = Logitech

2. Filtering

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”.

Pada contoh kasus ini kata kunci yang digunakan tidak ditemukan kata-kata yang tidak diperlukan, sehingga akan melanjutkan ke proses berikutnya yaitu *stemming*.

3. Stemming

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”.

Pada contoh kasus ini, kata kunci tidak mengandung imbuhan, sehingga akan melanjutkan ke proses berikutnya yaitu *term weighting*.

4. Term Weighting

Term Weighting adalah tahap dilakukannya penghitungan bobot tiap term yang dicari pada setiap data barang sehingga dapat diketahui ketersediaan dan kemiripan suatu term di dalam *database*.

Tabel 2. Hasil Pencarian

Kode Barang	Nama Barang	Stok Barang
B004	Keyboard Logitech	5
B006	Joystik Logitech	4
B001	Mouse Logitech	3
B003	Combo Mouse Keyboard Logitech	4
B002	Mouse Rexus	6
B005	Pointer Targus	7

Hasil Dan Pembahasan

Aplikasi pencarian barang pada Toko Mega Komputer Bengkulu merupakan suatu aplikasi yang digunakan untuk membantu pihak toko dalam mengelola barang yakni data barang dan penjualan barang. Pada form pencarian barang telah diterapkan suatu metode *rocchio relevance feedback* yang digunakan untuk memudahkan pencarian barang ketika ingin melakukan transaksi penjualan barang. Aplikasi ini dibuat menggunakan bahasa pemrograman *Visual Basic .Net* dan *Database SQL Server*.

Selain dapat melakukan pencarian barang, aplikasi ini juga dapat mencetak laporan penjualan per bulan, nota penjualan, serta dapat mencetak laporan persediaan barang di Toko Mega Komputer. Adapun hasil dari antarmuka aplikasi pencarian barang di Toko Mega Komputer, antara lain:

1. Login aplikasi

Login aplikasi merupakan *form* antarmuka pada aplikasi yang muncul pertama kali ketika membuka aplikasi. Pada *form login* aplikasi telah diberikan otentikasi

di *username* dan *password* sehingga jika memasukkan *username* atau *password* yang salah, maka sistem akan memberikan pesan status dari *login*. Adapun *form login* aplikasi.



Gambar 1. Menu Login Aplikasi

Menu utama merupakan *form* menu yang muncul ketika berhasil melakukan *login* pada aplikasi. Pada menu utama terdapat sub menu yang dapat diakses oleh pengguna, yaitu kategori barang, barang, pencarian barang, transaksi penjualan barang, nota penjualan, laporan persediaan barang, laporan penjualan barang per bulan.



Gambar 2. Form Menu Utama

Input data pengguna merupakan *form* input data yang digunakan untuk mengolah data

pengguna dari aplikasi pencarian barang pada Toko Mega Komputer. Proses yang dapat dilakukan yaitu menambah, mengubah, menghapus data pengguna.

kduser	username	pa
01	lian	140
USR01	admin	ad

Gambar 3. Form Input Data Pengguna

Input data kategori barang merupakan form input data yang digunakan untuk mengolah data kategori barang yang terdapat pada Toko Mega Komputer. Proses yang dapat dilakukan yaitu menambah, mengubah, menghapus data kategori barang

kdkategori	nmkategori
FL1	FLESDISH
K1	KEYBOARD
L1	LAPTOP
M1	MOUSE
P1	PRINTER
PC1	CPU
SP1	SPIKER

Gambar 4. Form Input Data Kategori Barang

Input data barang merupakan form input data yang digunakan untuk mengolah data barang yang terdapat pada Toko Mega

Komputer berdasarkan kategori barang. Proses yang dapat dilakukan yaitu menambah, mengubah, menghapus data barang.

Kategori Barang	kdbarang	nmbarang	qty	satuan	harga
CPU	PC1	PC AL IN ONE...	49	unit	4700
CPU	PC10	PC AIO LENO...	10	unit	4600
CPU	PC2	PC BUILD UP ...	10	unit	4650
CPU	PC3	PC BUILD UP ...	50	unit	5600
CPU	PC4	PC BUILD UP ...	30	unit	6500
CPU	PC5	PC MINI ACER...	60	unit	3950
CPU	PC7	PC BUILD UP ...	30	unit	7200

Gambar 5. Form Input Data Barang.

Input data transaksi penjualan barang merupakan form input data yang digunakan untuk mengolah data penjualan barang yang dibeli oleh konsumen. Proses yang dapat dilakukan yaitu menambah dan menghapus data penjualan barang. Penginputan data penjualan barang akan berpengaruh pada jumlah persediaan barang.

Kode Barang	Nama Barang	Harga Barang	Jumlah Penjualan	Total Harga
AA	ASUS E203	3350000	2	6700000
THF8	TOSHIBA 8 GB	95000	1	95000

Total Keseluruhan: 6795000
 Jumlah Bayar: 7000000
 Kembalian: 300000

yang digunakan untuk mencari barang yang akan dibeli oleh konsumen. Pada

form pencarian barang ini telah diterapkan metode Rocchio Relevance Feedback, sehingga hasil pencarian berdasarkan kata kunci dan nilai weighting yang didapat pada masing-masing kata kunci

Kode Barang	Nama Barang	qty	satuan	harga
AA	ASUS E203	1	unit	9350000
AB	ASUS A407MAH	6	unit	3850000
AC	ASUS A407M	8	unit	4100000
AD	ASUS X441M	6	unit	4250000
AE	ASUS X441M HD ...	4	unit	4500000
AG	ASUS X441UA	8	unit	5750000
AH	ASUS TP410U	5	unit	7950000
AI	ASUS XPP1UB	4	unit	6400000
AJ	ASUS X555Q	6	unit	7650000

Gambar 7. Form Pencarian Barang

Nota penjualan merupakan hasil dari proses pengolahan data penjualan barang dengan menampilkan list barang-barang yang dibeli berdasarkan nomor nota penjualan.

No	Kode Barang	Nama Barang	Harga Barang	Jumlah	Total Harga
1	AA	ASUS E203	Rp 3.350.000	2	Rp 6.700.000
2	THF8	TOSHIBA 8 GB	Rp 95.000	1	Rp 95.000
Total Keseluruhan					Rp 6.795.000

Gambar 8. Nota Penjualan

Laporan data persediaan barang merupakan hasil dari proses pengolahan data barang yang akan menampilkan jumlah persediaan.

No	Kode Barang	Nama Barang	Qty	Satuan	Total Harga
1	PC1	PC ALIN ONE ACER C20-720	49	unit	Rp 4.700.000
2	PC10	PC AIO LENOVO S200Z	10	unit	Rp 4.600.000
3	PC2	PC BUILD UP ACER XC-730	10	unit	Rp 4.650.000
4	PC3	PC BUILD UP ACER ATC-708	50	unit	Rp 5.600.000
5	PC4	PC BUILD UP ACER XC-780	30	unit	Rp 6.500.000
6	PC5	PC MINI ACER XC704G	60	unit	Rp 3.950.000
7	PC7	PC BUILD UP DELL 3470	30	unit	Rp 7.200.000
8	PC8	PC AIO DELL 3064	20	unit	Rp 7.200.000
9	PC9	PC AIO LENOVO V410Z	40	unit	Rp 7.600.000

No	Kode Barang	Nama Barang	Qty	Satuan	Total Harga
10	HPS	HP 8 GB	10	UNIT	Rp 120.000
11	kcf	kingston 4gb	28	unit	Rp 65.000
12	s48	sandisk extreme pro	2	unit	Rp 150.000
13	h4f4	toshiba 4gb	46	unit	Rp 75.000
14	THF8	TOSHIBA 8 GB	26	UNIT	Rp 95.000
15	lg	lg-gen usb titan 3.0 64gb	4	unit	Rp 250.000

Gambar 9. Laporan Data Persediaan Barang

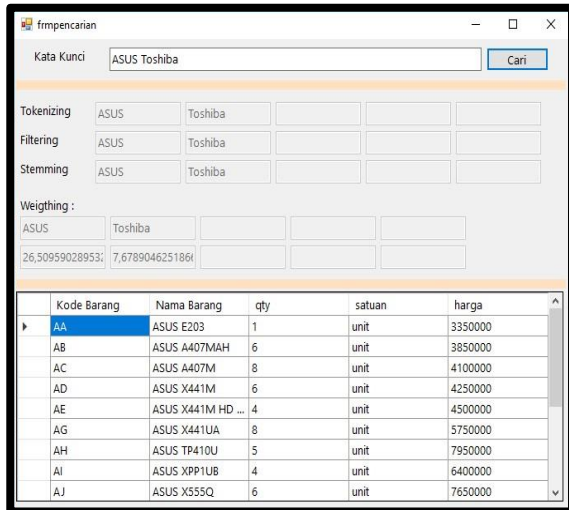
Laporan data Penjualan barang per bulan merupakan hasil dari proses pengolahan data transaksi penjualan barang yang dikelompokkan berdasarkan bulan yang dipilih.

No	No Nota	Kode Barang	Nama Barang	Harga Barang	Jumlah	Total Harga
1		A1	ACER A311-31	Rp 3.850.000	1	Rp 3.850.000
2		VT	VOTRE	Rp 25.000	1	Rp 25.000
3	001	A1	ACER A311-31	Rp 3.850.000	1	Rp 3.850.000
4	001	CP1	CANON PIXMA IP2770	Rp 670.000	3	Rp 2.010.000
5	001	h4f4	toshiba 4gb	Rp 75.000	4	Rp 300.000
6	002	SAV	ADVANCE	Rp 120.000	3	Rp 360.000
7	002	VT	VOTRE	Rp 25.000	2	Rp 50.000
8	003	VT	VOTRE	Rp 25.000	1	Rp 25.000
9	005	THF8	TOSHIBA 8 GB	Rp 95.000	2	Rp 190.000
10	005	VT	VOTRE	Rp 25.000	5	Rp 125.000
11	008	D1	DELL 3462	Rp 4.000.000	1	Rp 4.000.000
12	009	CP1	CANON PIXMA IP2770	Rp 670.000	2	Rp 1.340.000
13	010	CE10	EPSON LQ310	Rp 2.750.000	1	Rp 2.750.000
14	011	lg	logitech	Rp 25.000	3	Rp 75.000
15	012	A7	ACER ES-475G-59C7	Rp 7.650.000	1	Rp 7.650.000
16	013	SM	SEMBADA	Rp 450.000	1	Rp 450.000
17	014	D1	DELL 3462	Rp 4.000.000	1	Rp 4.000.000
18	014	kcf	kingston 4gb	Rp 65.000	2	Rp 130.000
19	015	A10	ACER Z476G	Rp 5.750.000	1	Rp 5.750.000

Gambar 10. Laporan Data Penjualan Barang Per Bulan

Rocchio memandang feedback sebagai permasalahan mencari sebuah query optimal, yaitu query yang memaksimalkan selisih antara data barang relevan dengan data barang tak relevan. Relevance feedback

bergun untuk mendekati *query* ke data barang yang relevan dan menjauhkan dari data barang tidak relevan. Ini bisa dilakukan melalui penambahan istilah *query* dan penyesuaian bobot istilah *query* sehingga sesuai dengan kegunaan istilah tersebut dalam fungsinya membedakan data barang relevan dan tak relevan.



Gambar 11. Pencarian Menggunakan Kata Kunci “ASUS Toshiba”

Langkah-langkah dalam pengolahan *query* adalah sebagai berikut :

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. Pada tahap ini, kata kunci akan diubah ke dalam bentuk *token* :

Token 1 = ASUS

Token 2 = Toshiba

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”. Pada contoh kasus ini kata kunci yang digunakan tidak ditemukan kata-kata yang tidak diperlukan, sehingga akan melanjutkan ke proses berikutnya yaitu *stemming*.

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”

Pada contoh kasus ini, kata kunci tidak mengandung imbuhan, sehingga akan melanjutkan ke proses berikutnya yaitu *term weighting*.

Term Weighting adalah tahap dilakukannya penghitungan bobot tiap term yang dicari pada setiap data barang sehingga dapat diketahui ketersediaan dan kemiripan suatu term di dalam database.

Token 1 = ASUS (26,5)

Token 2 = Toshiba (7,67)

Kesimpulan dan Saran

Kesimpulan

Berdasarkan hasil dari pembahasan, maka dapat disimpulkan bahwa :

1. Aplikasi pencarian barang pada Toko Mega Komputer Bengkulu merupakan suatu aplikasi yang digunakan untuk membantu pihak toko dalam mengelola barang yakni data barang dan penjualan barang. Pada form pencarian barang telah diterapkan suatu metode *rochio relevance feedback* yang digunakan untuk memudahkan pencarian barang ketika ingin melakukan transaksi penjualan barang
2. Aplikasi pencarian barang pada Toko Mega Komputer Bengkulu dibuat menggunakan bahasa pemrograman *Visual Basic .Net* dan *Database SQL Server*.
3. Pada aplikasi pencarian data barang batas kata yang diperbolehkan hanya
- 4 (empat) kata, dan pencarian data barang berdasarkan per token. Proses pencarian data barang akan melalui 4 (empat) tahapan yaitu *Tokenizing*, *Stemming*, *Filtering*, *Weighting*.
5. Berdasarkan hasil pengujian sistem, dapat disimpulkan aplikasi pencarian barang dapat membantu proses pengolahan data barang yang ada di Toko Mega Komputer.

Saran

1. Agar dapat menggunakan aplikasi pencarian barang dalam membantu mencari data barang yang diinginkan.

2. Perlu adanya pengembangan pada aplikasi ini, dengan cara menerapkan metode lain agar tidak perlu melewati proses pencarian dengan 4 fase tahapan.

Rocchio Relevance Feedback. Jurnal EECCIS Vol.7 No.1 Juni 2013.

Daftar Pustaka

- [1]Enterprise, Jubilee. 2015. *Pengenalan Visual Studio 2013*. PT. Elex Media Komputindo : Jakarta.
- [2]Farida, Naili. 2006. *Pentingnya Perlindungan Konsumen Dalam Pembelian Barang dan Jasa*. Majalah Pengembangan Ilmu Sosial Universitas Diponegoro Vol 34 No.2 Juni 2006. ISSN. 0126-0731.
- [3]Kusumo, Ario Suryo. 2016. *Administrasi SQL Server 2014*. PT. Elex MediaKomputindo : Jakarta.
- [4]Lubis, Adyanata. 2016. *Basis Data Dasar Untuk Mahasiswa Ilmu Komputer*. Penerbit Deepublish : Yogyakarta.
- [5]Merdekawan, Reyzak. Budianto, Alexius Endy. 2014. *Sistem Penelusuran Katalog Perpustakaan Menggunakan Metode Rocchio Relevance Feedback (Studi Kasus : Kantor Perpustakaan, Arsip dan Dokumentasi Kabupaten Tulungagung)*.
ejournal.unikama.ac.id
- [6]Mujilan, Agustinus. 2013. *Analisis dan Perancangan Sistem Perspektif Kompetensi Akuntansi Edisi 1*. Madiun. Universitas Widya Mandala.
- [7]Rosa. Shalahuddin. 2016. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Penerbit Informatika : Bandung.
- [8]Yugianus, Pausta. Dachlan, Harry Soekotjo. Hasanah, Rini Nur. 2013.
- [9]*Pengembangan Sistem Penelusuran Katalog Perpustakaan Dengan Metode*