Jurnal Ilmiah Widya Teknik
Volume 15 Nomor 2 2016
ISSN 1412-7350

SISTEM PEMEROLEHAN INFORMASI UNDANG-UNDANG DAN KASUS MENGGUNAKAN STRUKTUR DATA INVERTED INDEX DENGAN PEMBOBOTAN TF-IDF

Fredes Winda Oktaviani P.[1] , J.B. Budi Darmawan[2]
Universitas Sanata Dharma Yogyakarta, Jl. Affandi, Tromol Pos 29 Mrican Yogyakarta
Tlp. 0274-513301 Fax. 0274-562383
[1] windafredes04@gmail.com dan [2] b.darmawan@usd.ac.id

ABSTRAK

Sejak tahun 1945, pertambahan dokumen Undang-Undang dan kasus-kasus pelanggaran yang semakin banyak hingga saat ini menyebabkan sulitnya menemukan pasal-pasal yang sesuai dengan kasus pelanggaran yang ada maupun dokumen kasus-kasus yang mirip dengan kasus terbaru. Berdasarkan hal tersebut, maka diperlukannya suatu sistem Pemerolehan Informasi untuk memudahkan pencarian. Pada Pemerolehan Informasi, proses pencarian menggunakan struktur data inverted index dan pembobotan kata-kata menggunakan metode pembobotan TF-IDF. Umumnya sistem Pemerolehan Informasi mencari dokumen dengan konsep query to document, maka penulis ingin melakukan penelitian menggunakan konsep document to document dengan pencocokan kamus yang diambil dari isi tentang pada setiap Undang-Undang.

Percobaan ini dilakukan dengan menggunakan 100 dokumen kasus yang berisikan 40 term. Semua dokumen yang relevan dengan kata kunci yang berupa dokumen kasus dapat ditemukan kembali oleh sistem ini berdasarkan pencocokan kamus yang diambil dari isi tentang pada setiap Undang-Undang dengan operator OR. Nilai rata-rata precision yang diperoleh pada percobaan ini mencapai 0,65.

Kata kunci : Pemerolehan Informasi, TF-IDF, *Inverted Index*, *document to document*

I. Pendahuluan

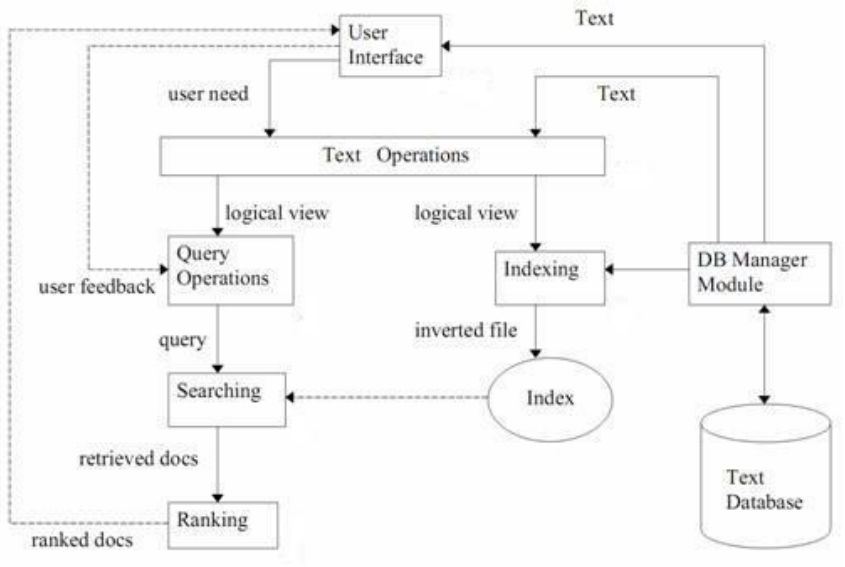
Sumber hukum dari Negara Indonesia terdapat pada Undang-Undang Dasar 1945 yang merupakan hukum dasar tertulis yang mengatur masalah kenegaraan dan merupakan dasar ketentuan-ketentuan lainnya yang harus ditaati. Tetapi pada kenyataanya sampai saat ini masih banyak kasus pelanggaran hukum yang terjadi baik di dalam rumah tangga, masyarakat, dan sebagainya. Misalnya saja dalam lingkup rumah tangga, yaitu pembatasan hak-hak anak dan kekerasan dalam rumah tangga, atau dalam lingkup masyarakat, yaitu korupsi, pencemaran nama baik dan lain-lain.

Dengan semakin banyaknya kasus pelanggaran tersebut, sehingga membuat para pihak yang berwenang (seperti advokat) yang membantu penggugat maupun tergugat merasa kesulitan karena harus mencari pasal-pasal pada kitab Undang-Undang. Hal tersebut akan memerlukan waktu yang sangat lama. Oleh sebab itu, dibutuhkannya suatu sistem yang dapat membantu pihak-pihak berwenang dalam memperoleh informasi pasal- pasal secara cepat dan tepat. Sistem yang akan digunakan ini memerlukan konsep Pemerolehan Informasi (Information Retrieval), yaitu aktivitas mendapatkan sumber informasi yang relevan untuk kebutuhan informasi dari suatu koleksi sumber informasi. Sistem pemerolehan informasi ini akan melakukan pencarian pasal-pasal pada Undang-Undang (1946-2014) yang terdapat pada dokumen yang tersimpan di basis data. Sistem ini akan menggunakan struktur data inverted index untuk mempercepat proses pencarian serta metode pembobotan TF- IDF untuk menemukan dokumen yang relevan dengan kasus pelanggaran. Selain itu, untuk membuat pemerolehan informasi ini menjadi efisien, diperlukan pula metode untuk mencari kata dasar dari term, yaitu stemming yang dibuat oleh Bobby Nazief dan Mirna Adriani[1], dan metode untuk menghilangkan term yang tidak terlalu penting, yaitu eliminasi stopwords. Kemudian setelah dilakukan proses pencarian, akan dilakukan evaluasi menggunakan metode recall dan precision untuk mengetahui keefektifan sistem dalam memperoleh undang-undang yang relevan.

II. Landasan Teori

II.1. Konsep Pemerolehan Informasi.

Pemerolehan Informasi merupakan proses yang terlibat dalam *representation*, *storage*, pencarian, dan mendapatkan informasi yang relevan untuk kebutuhan informasi yang diperlukan oleh pengguna[2]. Tipe dari informasi tersebut dapat berupa dokumen, halaman web, *online catalogs*, *structured records*, dan objek multimedia. Tujuan awal dari pemerolehan informasi ini adalah *indexing text* dan pencarian yang berguna pada suatu koleksi. Sekarang ini pemerolehan informasi telah melibatkan pemodelan, pencarian web, visualisasi data, penyaringan dan bahasa dalam memperoleh informasi yang relevan[3].Proses pemerolehan informasi disajikan pada Gambar 1.



Gambar 1. Proses Pemerolehan Informasi[3]

Dari Gambar 1 tersebut, terdapat 5 langkah utama dalam proses pemerolehan informasi. Tahap pertama yaitu operasi teks (*text operations*) contohnya proses eliminasi *stopwords* (penghilangan kata umum), proses *stemming* (pencarian kata dasar), dan sebagainya. Tahap kedua yaitu *query operations* contohnya penggunaan operator AND, OR, dan NOT pada *query*. Tahap ketiga yaitu pengindeksan (*indexing*) untuk mempercepat proses pencarian dimana *term* diindeksan dengan *id document*. Tahap keempat yaitu pencarian(*searching*) yang dilakukan pada *inverted file* yang sudah dibangun. Tahap kelima yaitu pembobotan(*ranking*) terhadap dokumen yang diperoleh dari proses pencarian.

II.2. Inverted Index

Inverted index atau *inverted file* merupakan struktur data pokok yang terdapat di sistem pemerolehan informasi[4]. *Inverted index* digunakan untuk mempercepat proses pencarian *terms* pada koleksi dokumen[5]. *Inverted index* memiliki dua komponen pokok yaitu *dictionary* dan *postings lists.* Untuk setiap *term* dalam koleksi, terdapat *posting list* yang mengandung informasi mengenai *term's occurences* di koleksi. Informasi yang ditemukan oleh *posting list* akan digunakan oleh sistem untuk memproses *query* pencarian[4].

II.2.1 Membangun Inverted Index

Tujuan dari membangun inverted index ini adalah untuk memperoleh kecepatan dalam pengindeksan saat melakukan proses pemerolehan (*retrieval*). Langkah utama dalam membangun *inverted index*[7] yaitu :

1. Kumpulkan dokumen-dokumen untuk diindekskan :

Friends, Romans, countrymen.	So let it be with Caesar	…

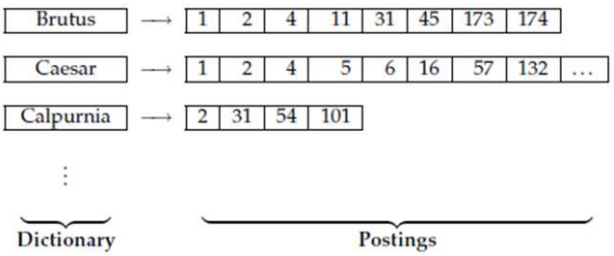
2. Lakukan proses *tokenization* pada text, kemudian kembalikan setiap dokumen ke dalam *list* :

Friends	Romans	countrymen	So	…

3. Lakukan proses berdasarkan ilmu bahasa, kemudian menghasilkan suatu *list* dari *tokens* yang sudah ternormalisasi, yang mengindekskan istilah-istilah :

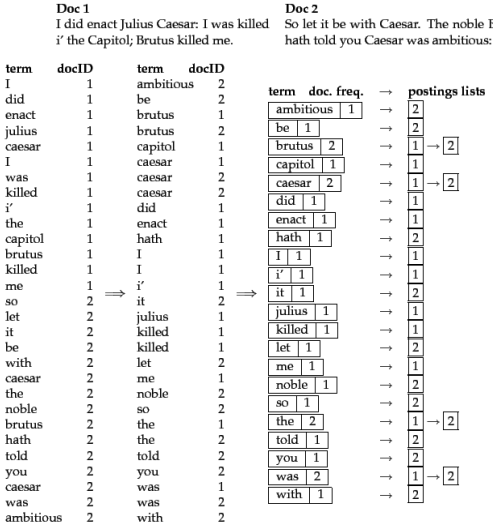
friend	roman	countryman	so	…

4. Lakukan pengindeksan terhadap dokumen-dokumen yang mengandung istilah-istilah tersebut dengan membuat *inverted index* yang terdiri dari *dictionary* dan *postings* yang direpresentasikan pada Gambar 2. berikut ini :



Gambar 2. *Inverted Index*[6]

Berikut ini pada Gambar 3 merupakan gambaran secara keseluruhan dari proses membangun *inverted index*[6]:



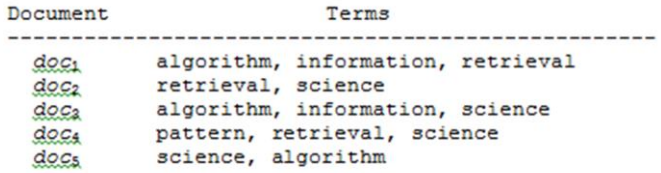
Gambar 3. Membangun *Inverted Index*

Pada Gambar 3 merupakan proses pembangunan *Inverted Index.* Pada sisi kiri dari gambar tersebut merupakan proses *tokenization* pada teks, kemudian pada bagian tengah merupakan proses pengurutan *term* dari proses *tokenization,* dan yang terakhir merupakan proses pengindeksan yang terdiri dari dictionary (kolom *term*) dan *postings lists.*

Dictionary berfungsi untuk menyimpan istilah-istilah dan mempunyai *pointer* untuk menuju ke *posting list* pada setiap istilah. Informasi (*term* dan *document frequency*) yang terdapat dalam *dictionary* dapat digunakan untuk meningkatkan efisiensi waktu *query* dan melakukan pembobotan pada model *ranked retrieval*. Sedangkan *posting list* berfungsi untuk menyimpan *list* dari dokumen yang mengandung suatu istilah tertentu. Selain itu, posting list juga dapat menyimpan informasi lain seperti frekuensi istilah atau posisi istilah atau posisi istilah dalam setiap dokumen[6].

II.2.2 Boolean Query pada Inverted Index

Pada proses pencarian ke struktur data *inverted index*, operasi yang digunakan adalah OR. Berikut ini pada merupakan gambaran umum dari proses operasi OR :



Gambar 4. Contoh Dokumen[7]

Terdapat query = "information or retrieval" dengan menggunakan contoh dokumen pada Gambar 5, maka hasilnya adalah : {*doc*1, *doc*3} ∪ {*doc*1, *doc*2, *doc*4} = {*doc*1, *doc*2, *doc*3, *doc*4}. Jadi semua dokumen yang mengandung kata "information" atau "retrieval" akan ditampilkan, yaitu doc1,doc2,doc3, dan doc4, sedangkan doc5 tidak ditampilkan karena tidak mengandung salah satu kata dari *query* tersebut.

II.3. Pembobotan TF-IDF

TF-IDF atau *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) merupakan dasar dari skema pembobotan istilah yang paling populer di pemerolehan informasi[3]. Teknik pembobotan TF-IDF menurut Savoy (1993)[8] :

$W_{ij} = ntf_{ij} * nidf_j$ … (1)

$ntf_{ij} = \frac{tfij}{Max\ tfi}$ dan $nidf_j = \frac{log\left(\frac{m}{dfj}\right)}{\log(m)}$ … (2)

Keterangan :

- W_{ij} adalah bobot istilah T_j pada dokumen D_i.
- tf_{ij} merupakan frekuensi dari istilah T_j dalam dokumen D_i.
- m adalah jumlah dokumen D_i pada kumpulan dokumen.
- df_j adalah jumlah dokumen yang mengandung istilah T_j.
- idf_j adalah kebalikan dari frekuensi dokumen (*inverse document frequency*)

- Max tf_i adalah frekuensi istilah terbesar pada satu dokumen

Pada teknik pembobotan TF-IDF ini, bobot istilah telah dinormalisasi, sehingga tidak perlu melakukan tahap normalisasi lagi. Penentuan bobot dari suatu istilah tidak hanya berdasarkan frekuensi kemunculan istilah pada satu dokumen, tetapi juga perlu memperhatikan frekuensi terbesar pada suatu istilah yang dimiliki oleh dokumen bersangkutan. Hal ini untuk menentukan posisi relatif bobot dari istilah dibanding dengan istilah-istilah lain di dokumen yang sama. Selain itu teknik pembobotan ini juga memperhitungkan jumlah dokumen. Hal ini berguna untuk mengetahui posisi relatif bobot istilah bersangkutan pada suatu dokumen dibandingkan dengan dokumen-dokumen lain yang memiliki istilah yang sama. Sehingga jika sebuah istilah mempunyai frekuensi kemunculan yang sama pada dua dokumen belum tentu mempunyai bobot yang sama[9].

II.4. Evaluasi Pemerolehan

Pada dasarnya ada dua pengukuran umum yang efektif, yaitu *recall* dan *precision* untuk membandingkan hasil pencarian. *Recall* digunakan untuk mengukur seberapa baik suatu sistem melakukan pencarian terhadap dokumen-dokumen yang relevan pada suatu *query*, sebaliknya *precision* digunakan untuk mengukur seberapa baik sistem tersebut menolak atau mengeliminasi dokumen-dokumen yang tidak relevan[10].

$$Recall = \frac{jumlah\ dokumen\ relevan\ yang\ diperoleh}{jumlah\ seluruh\ dokumen\ relevan} \ldots (3)$$

$$Precision = \frac{jumlah\ dokumen\ relevan\ yang\ diperoleh}{jumlah\ dokumen\ yang\ diperoleh} \ldots (4)$$

III. Metode Penelitian

III.1. Spesifikasi Software

Spesifikasi *software* yang digunakan dalam percobaan ini adalah :

1. Sistem operasi : Windows
2. MySQL Server 5.1
3. SQLyog 10.42
4. Java JDK 1.7.0
5. mysql-connector-java-5.1.6
6. Netbeans IDE 7.2

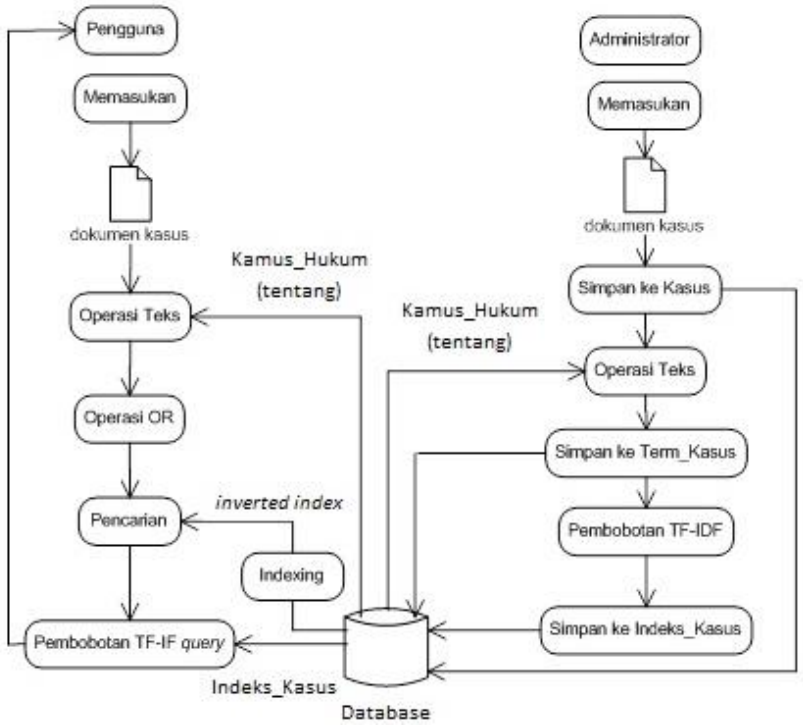
III.2. Prosedur Percobaan

Percobaan pada sistem ini dilakukan dengan memasukan dokumen kasus sebagai kata kunci pencarian. Kemudian penentuan relevansi dari output dokumen yang didapatkan. Dari hasil relevansi, recall dan precision dapat dihitung dengan mudah untuk menentukan interpolasi 11 titik. Langkah terakhir adalah menghitung rata-rata precision dari kelima percobaan.

III.3. Perancangan

Gambar 5 merupakan gambaran Sistem Pemerolehan Informasi Undang-Undang dan Kasus. Tahap pertama dari sisi pengguna pada Gambar 5 yaitu pengguna memasukan dokumen kasus (*.txt*) ke dalam sistem. Tahap kedua sistem akan melakukan operasi teks yaitu dengan melakukan pengambilan *term* yang sesuai dengan kamus_hukum dimana *term* yang berada di kamus_hukum diambil dari isi tentang pada setiap Undang-Undang. Hal ini bertujuan untuk mempercepat proses pencarian tanpa harus membandingkan dengan keseluruhan *term* yang ada di dokumen kasus. Tahap ketiga yaitu operasi teks dimana sistem akan mencari dokumen dengan menggunakan operator OR. Tahap keempat yaitu proses pencarian dimana sistem akan membandingkan dokumen kasus yang ada di struktur data *inverted index* yang telah dibangun. Tahap kelima yaitu pembobotan TF-IDF berdasarkan kata kunci dimana hasil dari pencarian dokumen yang didapat akan dibobotkan berdasarkan tingkat kesesuaian dari tabel Indeks_Kasus untuk mendapatkan hasil yang paling relevan dengan menggunakan pembobotan TF-IDF menurut Savoy.

Kemudian, untuk proses penambahan data kasus ke koleksi dokumen, tahap pertama yaitu pihak administrator memasukan dokumen-dokumen kasus baru ke dalam sistem beserta informasi yang diperlukannya, yaitu judul dan pasal yang bersangkutan. Tahap kedua, sistem menyimpan data tersebut ke tabel Kasus. Tahap ketiga, dokumen yang telah tersimpan di tabel Kasus akan disaring oleh sistem menggunakan operasi teks berdasarkan *term* di Kamus_hukum. Tahap keempat, hasil term yang sudah disaring tersebut akan disimpan ke tabel Term_Kasus. Tahap kelima, sistem melakukan pembobotan TF-IDF dengan menghitung dan menyimpan tf_ij, ntf_ij, dan w_ij ke tabel indeks_kasus serta df_j dan nidf_j ke tabel Term_Kasus.



Gambar 5. Perancangan

IV. Hasil Penelitian dan Pembahasan

Pada Tabel 1 berikut ini merupakan 5 contoh dari hasil percobaan yang telah dilakukan :

Tabel 1. Hasil Percobaan

Percobaan	Dokumen yang Diperoleh	Dokumen Sesuai dari yang Diperoleh	Dokumen Sesuai dari Seluruh Dokumen
I	16	5	5
II	8	6	6
III	8	8	8
IV	18	4	4
V	15	5	5

Berdasarkan Tabel 1 meskipun dokumen yang didapat tidak semuanya relevan dengan kata kunci, tetapi sistem dapat menemukan kembali semua dokumen yang relevan dengan menggunakan operasi OR.

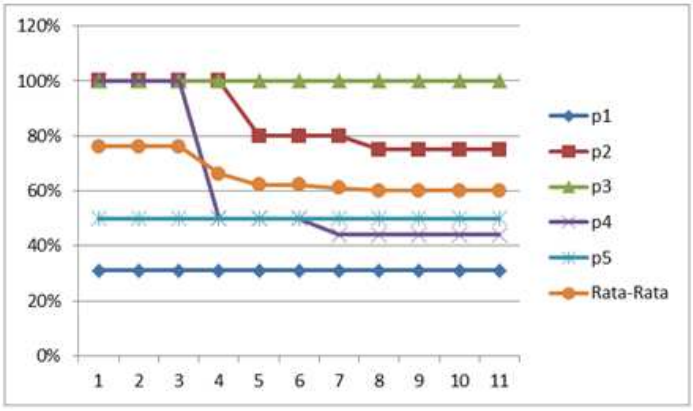
Pada Tabel 2 berikut ini merupakan hasil perhitungan rata-rata *precision* dari kelima percobaan tersebut :

Tabel 2. Rata-Rata *Precision*

Recall	**p1**	**p2**	**p3**	**p4**	**p5**	**Rata-Rata**
0%	31%	100%	100%	100%	50%	76%
10%	31%	100%	100%	100%	50%	76%
20%	31%	100%	100%	100%	50%	76%
30%	31%	100%	100%	50%	50%	66%
40%	31%	80%	100%	50%	50%	62%
50%	31%	80%	100%	50%	50%	62%
60%	31%	80%	100%	44%	50%	61%
70%	31%	75%	100%	44%	50%	60%
80%	31%	75%	100%	44%	50%	60%
90%	31%	75%	100%	44%	50%	60%
100%	31%	75%	100%	44%	50%	60%

Dari hasil rata-rata *precision* pada Tabel 1 dapat dihitung nilai rata-rata *precision* sebagai berikut :
(76%+76%+76%+66%+62%+62%+61%+60%+60%+60%+60%)/11 = 65% = 0.65

Pada Gambar 6 di bawah ini merupakan rata-rata dari kelima *precision* yang digambarkan dalam bentuk diagram garis :



Gambar 6. Rata-Rata *Precision*

Berdasarkan Gambar 6 terlihat grafik nilai rata-rata *precision* membentuk garis ke arah kanan atas hampir menyerupai garis horizontal. Semakin garis tersebut mengarah ke kanan atas hingga membentuk garis horizontal dan nilai *precision* mendekati 1, maka tingkat relevansi atau kecocokan dokumen yang dihasilkan dari suatu sistem akan semakin baik. Jadi, nilai *precision* berdasarkan perhitungan rata-rata *precision* pada Sistem Pemerolehan Informasi Undang-Undang dan Kasus ini adalah 0.65.

V. Kesimpulan

Dengan percobaan menggunakan 100 dokumen kasus yang berisikan 40 *term*, Sistem Pemerolehan Informasi Undang-Undang dan Kasus ini dapat menemukan kembali semua dokumen yang relevan dengan masukan berupa dokumen kasus untuk menghasilkan kata kunci berdasarkan pencocokan kamus yang diambil dari isi tentang pada setiap Undang-Undang dengan operator OR. Nilai rata-rata *precision* yang diperoleh pada percobaan ini mencapai 0,65.

Daftar Pustaka

1. Agusta, Ledy, 2009, “Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani untuk Stemming Dokumen Teks Bahasa Indonesia”. Jurnal Konferensi Nasional Sistem dan Informatika 2009, Bali.
2. Ingwersen, Peter, 2002, “Information Retrieval Interaction”. Denmark: Royal School of Library and Information Science.
3. Baeza-Yates, R., Ribeiro-Neto, B., 1999, “Modern Information Retrieval the Concepts and Technology Behind Search”. England: A division of the association for Computing Machinary.
4. Büttcher, Stefan., Clarke, L.A Charles., Cormack, V. Gordon, 2010, “Information Retrieval Implementing and Evaluating Search Engines”. USA: Massachusetts Institute of Technology.
5. Grossman, David A., & Frieder, Ophir, 2004, “Information Retrieval Alhorithms and Heuristics”. USA: Illinois Institute of Technology Chicago.
6. Manning, Christopher, D., Raghavan, Prabhakar., Schütze, Hinrich, 2008, “Introduction to Information Retrieval”. England: Cambridge University Press.
7. Netmeh, Brian A., Dickey, Thomas E., Wartik, Steven P., 1989, “Traceability Technology at the Software Productivity Consortium”.
8. Savoy, J., 1993, “A Learning Scheme for Information Retrieval in Hypertext”. Information Processing & Management, 30(4), 515-533.
9. Hasibuan, Zainal A., & Andri, Yofi, 2001, “Penerapan Berbagai Teknik Sistem Temu-Kembali Informasi Berbasis Hiperteks”. Jurnal Ilmu Komputer dan Teknologi Informasi, Volume 1, Nomor 2.
10. Croft, Bruce W., Metzler, Donald., Strohman, Trevor, 2010, “Search Engines Information Retrieval in Practice”. USA: University of Massachusetts, Amherst. Pearson Education USA.