Implementasi Metode Document Oriented Index Pruning pada Information Retrieval System

Hendri Priyambowo[1], Yanuar Firdaus A.W. S.T, M.T[2], Siti Sa'adah S.T. M.T[3]

[123]Program Studi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom

Abstrak - Perkembangan Informasi yang sangat pesat mengakibatkan jumlah informasi yang tersedia secara online mengalami peningkatan yang sangat pesat, sehingga sangat sulit apabila pencarian dilakukan secara satu-persatu, karenanya dibutuhkan *Information Retrieval System* untuk menemukan suatu informasi.

Permasalahan yang muncul pada *Information Retrieval* adalah semakin besar *data collection* yang dimiliki maka semakin besar pula biaya yang dibutuhkan untuk menyediakan layanan komputasi, *storage,* dan *network resource.* Oleh karena itu suatu metode diperlukan untuk mengatasi permasalahan tersebut yaitu dengan kompresi *index*. Metode kompresi *index* yang akan digunakan adalah metode *Document Oriented Index Pruning.*

Berdasarkan penelitian yang telah dilakukan sampai dengan persentase *index* sebesar 80% metode *Document Oriented Index Pruning* mampu memberikan hasil relevansi pencarian yang lebih baik dibandingkan hasil relevansi pencarian tanpa menggunakan metode *Document Oriented Index Pruning* sehingga performansi sistem yang dihasilkan lebih baik dibandingkan dengan tanpa menggunakan metode *Document Oriented Index Pruning.*

Kata kunci:*information retrieval , document oriented index pruning, indexing*

1. Pendahuluan

Perkembangan Informasi yang sangat pesat mengakibatkan jumlah informasi yang tersedia secara online mengalami peningkatan yang sangat pesat, sehingga sangat sulit apabila pencarian dilakukan secara satu-persatu, karenanya dibutuhkan *Information Retrieval System* untuk menemukan suatu informasi. *Information Retrieval* bisa dikatakan sebagai teknik untuk mencari sesuatu(informasi) yang berada pada *data collection* yang berukuran besar dan sesuai dengan apa yang kita inginkan [1].

Secara umum proses yang terjadi di *Information Retrieval* adalah *indexing* dan *searching.* Pada *indexing* dibentuk struktur data yang akan digunakan untuk menemukan informasi, sementara pada *searching* dilakukan pencocokan *query* masukan *user* dengan data yang ada pada struktur data(*index*).

Permasalahan yang muncul pada *Information Retrieval* adalah semakin besar *data collection* yang dimiliki maka semakin besar pula biaya yang dibutuhkan untuk menyediakan layanan komputasi, *storage,* dan *network resource* [2]. Salah satu solusi atas permasalahan diatas adalah dengan mengurangi ukuran *inverted index. Inverted index* adalah struktur data yang digunakan pada *Information Retrieval* (IR). Ada beberapa metode yang dapat digunakan yaitu *posting oriented pruning, term oriented pruning,* dan *document oriented index pruning.*

Metode yang akan digunakan pada tugas akhir ini adalah metode *Document Oriented Index Pruning*. Metode ini dipilih karena metode ini mampu menjaga relevansi hasil pencarian lebih baik daripada metode lainnya. Ide dasar dari metode ini adalah dengan mengeliminasi seluruh kolom dari *index table*, berdasarkan asumsi tidak semua *document* yang ada di *collection document* itu penting. Pemilihan *document* yang akan dieliminasi berdasarkan *importance Score* dari masing-masing document [2].

Tugas akhir ini akan berfokus pada kompresi *index* pada *Information Retrieval* dan kemudian dilakukan analisa performansi sistem pada beberapa persentase ukuran *index* setelah dilakukan pemotongan *index*. Analisa performansi sistem akan dilakukan pada tahap *indexing* dan *searching*. Parameter performansi yang akan digunakan pada proses *indexing* adalah *index size* dan *index construction time*(waktu pembangunan *index*), sementara pada proses *searching* parameter yang akan digunakan adalah *precision*, *recall*, dan *query time*.

2. Landasan Teori

Document Oriented Index Pruning

Document Oriented Index Pruning adalah metode kompresi *Index* dengan cara menghilangkan sebagian document yang tidak diperlukan. Ide dasar dari metode ini adalah dengan menghilangkan kolom yang ada pada *index table*. Pemilihan document mana yang akan dihilangkan dilakukan berdasarkan *importance score*-nya.

Pada metode ini diasumsikan bahwa *document* yang mengandung *term* yang sering muncul tidak lebih penting dibandingkan dengan *document* yang mengandung *term* yang jarang muncul [2].

persamaan untuk menentukan importance score :

$$w(d) = \frac{1}{|d|} \sum_{d} tf \quad (t).NI$$

$$w(d) = \frac{1}{|d|} \sum_{d} tf \quad (t).\log\left(\frac{N - N_t + 0{,}5}{N_t + 0{,}5}\right)$$

Keterangan :

t_i = nilai term frequency term i dari dokumen d

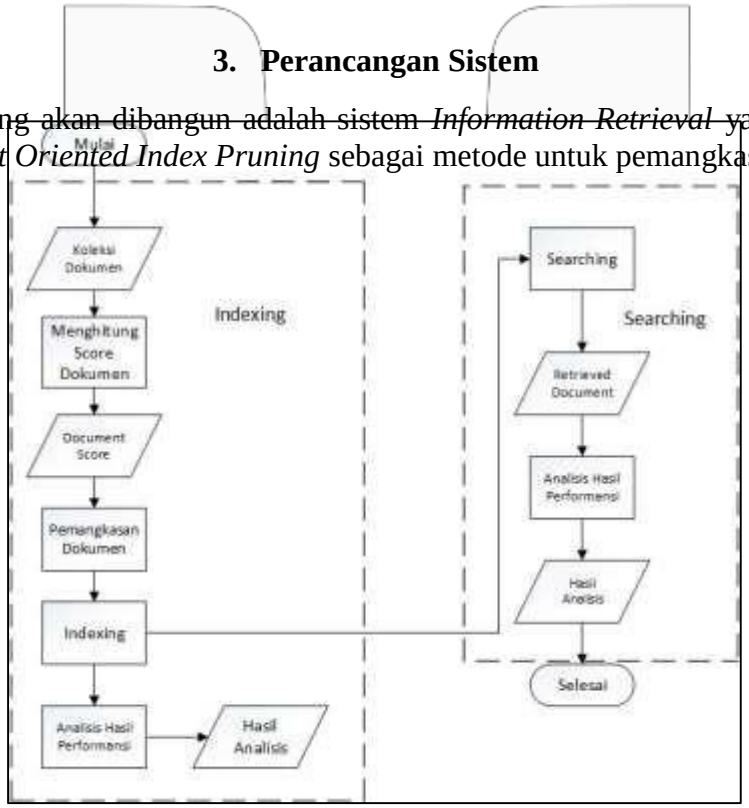
N = jumlah keseluruhan dokumen

N_{ti} = jumlah kemunculan term i di seluruh dokumen

|d| = denominator berupa panjang dokumen

3. Perancangan Sistem

Sistem yang akan dibangun adalah sistem *Information Retrieval* yang menggunakan metode *Document Oriented Index Pruning* sebagai metode untuk pemangkasan *index.*



Berdasarkan Gambaran Umum Sistem yang dibangun proses yang akan dilakukan adalah

a. Sistem menerima inputan berupa file koleksi dokumen dengan format txt. Dari file koleksi dokumen tersebut dilakukan pemisahan berdasarkan token-token yang ada di dalam masing masing dokumen berdasarkan id dokumen.
b. Dari koleksi dokumen yang dimiliki dilakukan *scoring document* menggunakan metode *Document Oriented Index Pruning*, untuk mendapatkan nilai *importance score* dari tiap-tiap dokumen.
c. Setelah itu dokumen diurutkan berdasarkan importance score dari tiap-tiap dokumen.
d. Kemudian dilakukan pemangkasan dokumen berdasarkan persentase *index* yang akan digunakan pada sistem. Pemangkasan dokumen dilakukan berdasarkan nilai *importance score* dari dokumen.
e. Setelah itu dilakukan *indexing* untuk membentuk struktur data dari sistem, kemudian dilakukan analisis hasil performansi.
f. Kemudian dilakukan proses *searching* dari *index* yang telah terbentuk, *searching* dilakukan berdasarkan *query* yang telah ditentukan.
g. Hasilnya kemudian di analisis.

4. Pengujian dan Analisis

Skenario Pengujian

Pengujian sistem yang telah dibangun dilakukan dengan menggunakan *query* yang telah ditentukan, *query* tersebut diperoleh dari http://ir.dcs.gla.ac.uk/resources/test_collections/ yang terdiri dari 40 *query*. *Query* yang digunakan adalah *query* yang memiliki daftar *relevant judgement* yang lebih besar dari 10 buah untuk mendapatkan kemungkinan nilai *precision* dan *recall* yang tidak terlalu berbanding jauh untuk setiap persentase *index* yang digunakan. Pengujian akan dilakukan dengan membandingkan hasil pengujian berdasarkan ukuran *index*. Ukuran *index* yang akan digunakan adalah 100%, 95%, 90%, 85%, 80%, 75%, 70%, 65%, 60%, 55%, 50%,45%, 40%, 35%, 30% dari ukuran *index* awal. Pengujian dilakukan untuk mendapatkan nilai *indexsize*, *querytime*, *precision* dan *recall*. Nilai-nilai tersebut akan dibuat grafik kemudian dilakukan analisis.

Pada tahapan *Searching* akan digunakan *query* sebanyak 40 *query* dan untuk menentukan nilai parameter performansi sistem dari total keseluruhan *query* akan diambil nilai rata-rata untuk setiap persentase *index*.

Hasil Pengujian

A. Hasil pengujian tahapan indexing

Tabel 1 Hasil performansi tahap Indexing

Ukuran Index	Index Size(KB)	Index Construction Time(detik)
100%	10828,32	11,24
95%	10426,096	10,161
90%	9975,592	10,375
85%	9501,408	9,555
80%	9008,472	8,957
75%	8521,16	8,25
70%	8021,016	7,264
65%	7500,208	6,71
60%	6940,432	5,966
55%	6375,968	5,29
50%	5782,024	4,524
45%	5173,336	4,19
40%	4554,432	3,536
35%	3909,232	2,983
30%	3293,296	2,705

Dari hasil yang diperoleh pada Tabel 1 terlihat bahwa untuk setiap persentase pemotongan *index* didapatkan nilai *index construction time* serta nilai *index size* yang berbeda-beda. Nilai *index size* serta *index construction time* terbesar diperoleh pada persentase *index* 100%, sementara nilai index size serta *index construction time* terkecil diperoleh pada persentase *index* 30%.

B. Hasil pengujian tahapan searching

Tabel 2 Hasil performansi tahapan Searching

Ukuran Index	Query Time(s)	Recall	Precision
100%	0,074	0,76	0,037
95%	0,07	0,71	0,037
90%	0,066	0,68	0,037
85%	0,062	0,63	0,037
80%	0,059	0,59	0,037
75%	0,0551	0,55	0,036
70%	0,051	0,5	0,036
65%	0,048	0,47	0,037
60%	0,044	0,43	0,037
55%	0,041	0,38	0,034
50%	0,037	0,33	0,032
45%	0,034	0,29	0,032
40%	0,03	0,25	0,032
35%	0,026	0,21	0,031
30%	0,023	0,17	0,03

Dari hasil yang diperoleh pada Tabel 2 terlihat bahwa untuk nilai *recall* serta nilai *query time* yang dihasilkan untuk setiap persentase *index* yang digunakan diperoleh nilai yang berbeda-beda. Nilai *query time* serta nilai *recall* tertinggi diperoleh pada persentase *index* 100% sementara nilai *query time* serta nilai *recall* terendah diperoleh pada persentase *index* 30%. Untuk nilai *precision* untuk beberapa persentase *index* diperoleh nilai yang sama, misalnya dapat dilihat pada tabel 2 untuk persentase *index* 100% sampai dengan 80% diperoleh nilai *precision* yang sama.

5. Kesimpulan dan Saran

Berdasarkan hasil pengujian dan analisis yang dihasilkan selama pengerjaan tugas akhir ini, dapat disimpulkan bahwa :

1. Proses penerapan metode *Document Oriented Index Pruning* dapat dilakukan dengan melakukan proses *scoring document* kemudian hasil *scoring* diurutkan berdasarkan *importance score* tiap-tiap dokumen dan penentuan dokumen yang dihilangkan dilakukan berdasarkan *importance score*.
2. Pada proses *indexing* untuk setiap pemangkasan 5% index terjadi pengurangan 400 – 500 KB *index size* dan 0.2 sampai 1 detik *index construction time* sehingga semakin kecil persentase *index* yang digunakan maka semakin kecil pula nilai *index size* serta nilai *index construction time* atau waktu yang dibutuhkan untuk membentuk *index* yang dihasilkan.
3. Pada proses searching metode *Document Oriented Index Pruning* dapat meningkatkan performansi pada persentase *index* sampai dengan 80%. Kesimpulan ini didasarkan dari semakin rendahnya nilai *query time*yaitu terjadi penurunan nilai *query time* sebesar 0.003 – 0.005 detik untuk setiap 5% pemangkasan index yang dihasilkan dengan tetap menghasilkan nilai *precision* yang sama baiknya dengan persentase *index* 100% atau ukuran *index* asli.

Adapun saran yang dapat digunakan, jika ingin melanjutkan penelitian ini :

1. Menggunakan varian pembobotan yang lainnya seperti *augmented*, *bolean* ,*pivoted unique weighting* atau varian lainnya,
2. Membandingkan performansi antara metode-metode *pruning index* yang ada pada *Information Retrieval* seperti metode *posting oriented pruning* atau *term oriented pruning* atau mencoba mengkombinasikan algoritma-algoritma tersebut.

Referensi

[1] C. D. Manning, R. Prabhakar and H. Schütze, An Introduction to Information Retrieval, Cambridge: Cambridge University Press, 2009.

[2] L. Zheng and C. I. J, "Document Oriented Pruning of The Inverted Index in Information Retrieval Systems," *International Conference on Advanced Information Networking ang Aplications Workshop,* pp. 697-702, 2009.

[3] C. J. v. RIJSBERGEN, Information Retrieval, Glasgow: Information Retrieval Group , 1979.

[4] A. Silberschatz, H. F. Korth and S. Sudarshan, Database System Concept, McGraw-Hill , 2005.

[5] Bernardi, "Department of Information Engineering and Computer Science University of Toronto," [Online]. Available: http://disi.unitn.it/~bernardi/Courses/DL/Slides_11_12/measures.pdf. [Accessed 19 March 2015].

[6] A. Z. Broder, N. Eiron, M. Fontoura, M. Herscovici, R. Lempel, J. McPherson, R. Qi and E. Shekita, "Indexing Shared Content in Information".

[7] J. Prasetyo, "IMPLEMENTASI DAN ANALISIS METODE STATIC INDEX PRUNING PADA INFORMATION RETRIEVAL SYSTEM UNTUK MENINGKATKAN PERFORMANSI SEARCH ENGINE," Telkom University, Bandung, 2011.

[8] Mausam, "Document Similarity in Information Retrieval," [Online]. Available: https://courses.cs.washington.edu/courses/cse573/12sp/lectures/17-ir.pdf. [Accessed 17 March 2015].

[9] "UCI Donald Bren School of Information & Computer Science," [Online]. Available: http://www.ics.uci.edu/~djp3/classes/2008_09_26_CS221/Lectures/Lecture26.pdf. [Accessed 17 March 2015].

[10] B. Zhou and Y. Yao, "Evaluating Information Retrieval System Performance".

[11] R. B. González, Index Compression for Information Retrieval Systems, 2008.

[12] A. Singhal, C. Buckley and M. Mitra, "Pivoted Document Length Normalization".

[13] "Clemson University," 2002. [Online]. Available: http://people.cs.clemson.edu/~juan/CPSC862/Concept-50/IR-Basics-of-Inverted-Index.pdf. [Accessed 18 March 2015].

[14] J. D. Anderson, "Guidelines for Indexes and Related Information Retrieval Devices," *The National Information Standards Organization,* 1997.

[15] "Database Index," [Online]. Available: https://en.wikipedia.org/wiki/Database_index. [Accessed 3 March 2016].

[16] "Search Engine Indexing," [Online]. Available: https://en.wikipedia.org/wiki/Search_engine_indexing. [Accessed 3 March 2016].