MESIN PENCARI BERBASISKAN SEMANTIK UNTUK BAHASA INDONESIA

Putu Wuri Handayani[1], I Made Wiryana[2], dan Jan-Torsten Milde[3]

[1]Fakultas Ilmu Komputer, Universitas Indonesia, Depok, Indonesia
putu.wuri@cs.ui.ac.id
[2]Universitas Gunadarma
mwiryana@gmail.com
[3]University of Applied Science Fulda
milde@hs-fulde.de

Abstrak

Makin meningkatnya jumlah informasi yang terdapat di Internet menjadi salah satu alasan yang kuat untuk mengembangkan mesin pencari yang handal. Google merupakan salah satu mesin pencari yang handal namun masih memiliki keterbatasan khususnya dalam melakukan analisa kandungan dokumen. Tujuan dari penelitian ini adalah untuk mengembangkan mesin pencari yang dapat menganalisa kandungan teks bahasa Indonesia dengan menggunakan metodologi studi literatur dan penelitian lapangan. Berdasarkan hasil uji coba, penggunaan analisa semantik mempermudah pengguna dalam mencari artikel yang dibutuhkan.

Kata kunci : *search engine*, *tag of cloud*, *semantic analysis*

1. Pendahuluan

Saat ini, sebagian besar orang menggunakan mesin pencari (*search engine*) untuk mencari berbagai informasi di internet. Oleh karena itu, mesin pencari yang handal sangat dibutuhkan untuk memberikan informasi yang tepat dan akurat. Salah satu contoh mesin pencari yang handal dan paling banyak digunakan oleh pengguna di Indonesia saat ini adalah Google Indonesia [www.google.co.id]. Google Indonesia banyak dipilih oleh para pengguna karena memiliki *User Interface* yang sederhana dan dapat mencari di banyak URL. Namun, Google Indonesia masih memiliki keterbatasan, terutama untuk dokumen dalam bahasa Indonesia.

Berawal dari kebutuhan mesin pencari di dalam situs Presiden Republik Indonesia [http://www.presidensby.info], dan setelah dilakukan analisa perbandingan terhadap beberapa mesin pencari yang ada, maka dikembangkanlah sebuah sistem mesin pencari yang cukup memahami bahasa Indonesia dengan melakukan analisa kandungan dari kalimat, contohnya adalah ketika pengguna mengetikkan kata kunci "beruang", informasi yang dihasilkan oleh mesin pencari adalah artikel mengenai hewan beruang dan atau orang yang kaya raya.

Beberapa mesin pencari telah mampu melakukan identifikasi bahasa yang digunakan pada dokumen tersebut. Proses identifikasi ini biasanya dilakukan dengan cara mengenali beberapa kata di dokumen tersebut yang merupakan ciri atau kekhasan bagi bahasa tertentu, akan tetapi mesin pencari tersebut tidak melakukan analisa terhadap kandungan dari dokumen tersebut. Akibatnya, untuk beberapa kondisi pencarian menjadi sangat terbatas dan bahkan menyebabkan pencarian yang memberikan hasil yang tidak memiliki hubungan dengan makna kata yang ingin ditemukan.

Berdasarkan penjelasan diatas, maka masalah utama yang dapat diidentifikasi di dalam makalah ini adalah:

a. Sebagian besar mesin pencari yang ada masih memberikan hasil yang tidak sesuai dengan kata kunci yang dimasukkan.
b. Perlunya metode akses untuk pencarian ulang dalam mencari informasi yang tepat sesuai dengan konteks kata kunci yang dimasukkan.
c. Sangat sulit untuk menemukan hasil yang sesuai dengan konteks kata kunci karena kurangnya bentuk formal tata bahasa Indonesia

2. Tinjauan Pustaka

Dalam membangun mesin pencari ini berkaitan erat dengan *Natural Language Processing* (NLP) dan *semantic web*. "*Natural Language Processing is a range of computational techniques to analyze and represent naturally occurring text (free text) at one or more levels of linguistic analysis (e.g. phonological, morphological, lexical, syntactic, semantic, discourse, pragmatic) for the purpose of achieving*

human-like language processing for knowledge-intensive applications [1].” Penelitian pertama yang berkaitan dengan *Natural Language Processing* dilakukan oleh Roger C. Schank dalam membuat parser otomatis untuk *natural language* [2]. Dan belum lama ini, Microsoft Research Group telah melalukan penelitian untuk mengembangkan piranti lunak yang dapat mengalisa kandungan teks dalam suatu artikel [3].

Dalam menganalisa kandungan teks, teknik *parsing* dan tata bahasa sangat diperlukan untuk memeriksa struktur sintaks dari suatu kalimat yang akan dianalisa. Setelah struktur sintaks dari suatu kalimat sudah dapat diidentifikasi, maka subjek, predikat dan objek dari kalimat tersebut dapat didefinisikan. Penentuan subjek, predikat dan objek tersebut bertujuan untuk menemukan *tag-tag* yang mungkin dihasilkan dari kalimat tersebut. *Parsing* juga dapat dilakukan dengan menggunakan *regular expression*. *Regular expression* ini memerlukan *pattern* dan *corpus* untuk mencari kata-kata dalam kalimat.

Teknologi *semantic web* digunakan dalam penelitian ini untuk mengelola informasi, mendefinisikan data semantik dan data semantik tersebut akan digunakan untuk menganotasikan teks dalam suatu artikel. Sebagai contoh, apabila pengguna mencari artikel yang berhubungan dengan “Presiden RI” maka mesin pencari juga akan dapat menampilkan artikel yang berkaitan dengan “Susilo Bambang Yudhoyono” yang merupakan Presiden RI yang menjabat pada saat ini. Dalam membuat anotasi semantik maka diperlukan suatu ontologi yang mendefinisikan konsep atau arti dalam suatu domain pengetahuan.

3. Metode Penelitian

Metodologi yang digunakan dalam pengembangan mesin pencari ini adalah studi literatur dan penelitian lapangan. Studi literatur diambil dari buku, jurnal penelitian, artikel dan dokumentasi yang berkaitan dengan *Natural Language Processing* dan *semantic web*. Sedangkan penelitian lapangan menggunakan metode observasi dan survei dengan kuesioner yang dilakukan secara *online* untuk mengetahui apakah mesin pencari ini dapat lebih membantu pengguna menemukan artikel dalam Bahasa Indonesia dibandingkan dengan mesin pencari yang sudah ada.

4. Hasil dan Pembahasan

Untuk menyelesaikan masalah utama diatas, maka purwa rupa mesin pencari yang dikembangkan ini menggunakan dua pendekatan yang masih jarang digunakan oleh mesin pencari lainnya. Pertama, dalam melakukan proses pencarian yaitu menggunakan analisa kandungan teks Bahasa Indonesia. Kedua, dalam penyajian hasil pencarian serta metoda memfokuskan pencarian yang menggunakan "*cloud of tag*" atau kumpulan *tag*. *Tag* adalah kata atau frase yang digunakan untuk merepresentasikan topik dari suatu artikel. *Tag* ini dihasilkan dengan memanfaatkan *topic map* yang disusun berdasarkan ontologi yang telah didefinisikan sebelumnya, sebagai contoh ontologi bahasa dapat memiliki *topic map* bahasa Inggris, bahasa Jerman, bahasa Belanda, dan sebagainya. Selain itu, masing-masing *tag* memiliki bobot nilai sesuai dengan nilai relevansi *tag* tersebut dengan kata kunci yang diketikkan oleh pengguna.

Pemanfaatan *tag* ini bertujuan untuk mempermudah pengguna ketika melakukan proses pencarian. Pengguna cukup memilih topik yang berkaitan dengan kata kunci yang pertama digunakan untuk memfokuskan hasil pencarian. Misalkan, kata kunci ”Bali” memiliki *cloud of tag* ”Pulau Bali”, ”Tari Bali”, ”Orang Bali” dan lain-lain. Setelah itu pengguna dapat mempersempit pencarian dengan memilih topik yang dikehendaki. Keuntungan lain dari penggunaan *tag* adalah untuk mengurangi berbagai macam *tag* yang memiliki arti yang sama didalam sebuah artikel, sebagai contoh *tag* ”Presiden RI” dan ”Kepala Negara RI” dapat diminimalkan menjadi satu *tag* yaitu ”Presiden RI” saja. Selain itu, *tag* juga dapat digunakan untuk membantu pengguna dalam menggeneralisasikan informasi umum ke dalam informasi yang lebih khusus.

Penjelasan diatas dapat dituangkan dalam suatu formula rumus matematikanya:

Definisi 1

S = {A1, A2, ..., An} dimana S = Sistem, A = Artikel

Definisi 2

A = {T1, T2, …, Tn}dimana T = *Tag*

Definisi 3

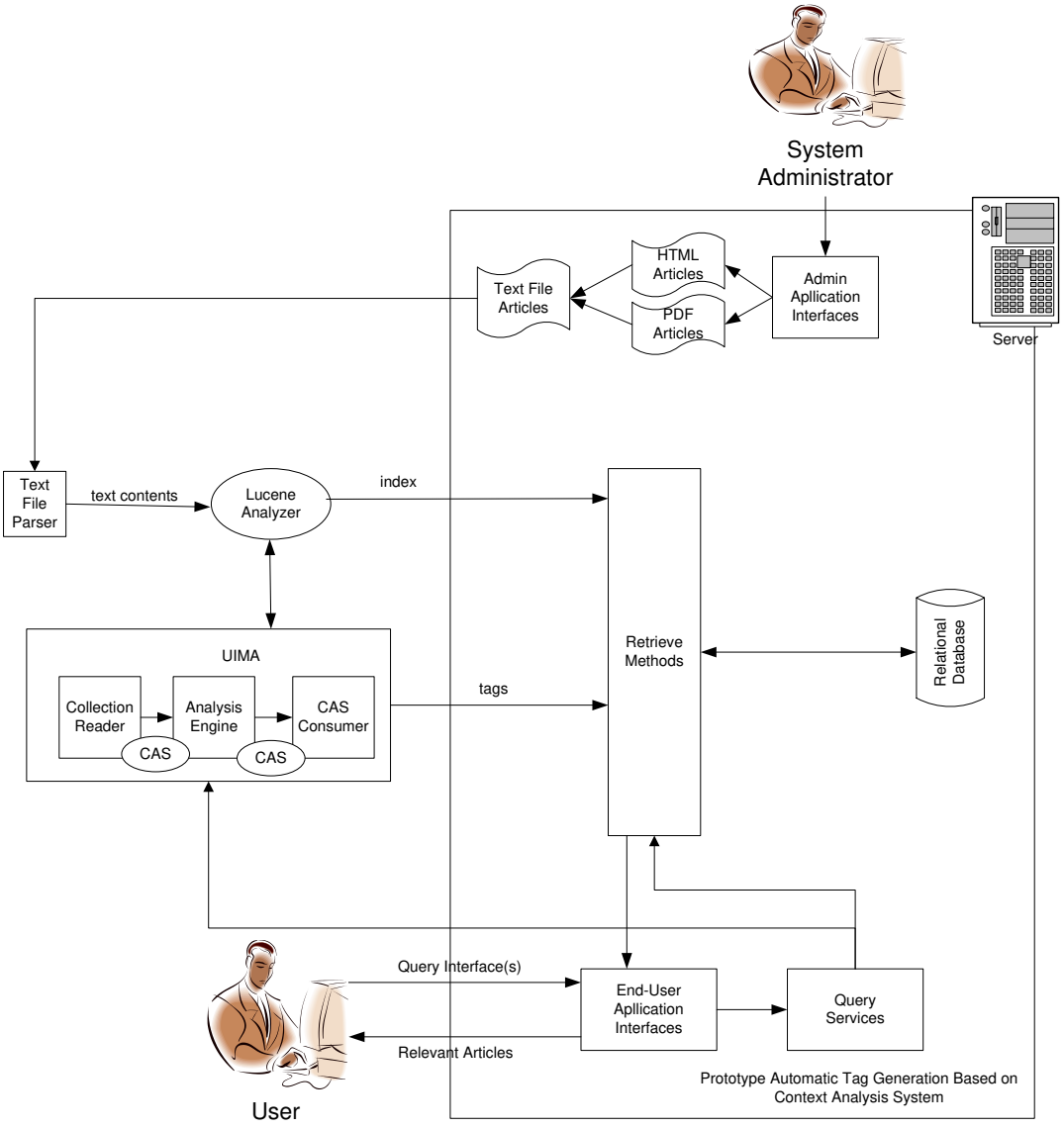
T = <w,s> dimana w = kata, s = bobot nilai

Definisi 4

CT = T1 U T2 U … U Tn, dimana CT = *Cloud of Tag*

Pada purwa rupa mesin pencari ini, *tag* yang memiliki bobot nilai relevansi yang tinggi sesuai dengan konteks kata kunci yang dimasukkan oleh pengguna akan direpresentasikan dengan ukuran *font* yang paling besar dan begitu juga sebaliknya.

Untuk mengantisipasi penambahan informasi yang semakin pesat di masa depan dan sesuai dengan kecenderungan perkembangan *web* dimana pengguna dapat ikut aktif, maka dalam purwa rupa mesin pencari ini disediakan fitur untuk menambah *tag* yang dapat digunakan langsung oleh pengguna. Dengan



Gambar 1. Sistem Arsitektur *Prototype* Mesin Pencari

kata lain, pengguna dapat menambahkan *topic map* yang berkaitan dengan kata kunci yang dimasukkan.

Sistem mesin pencari ini berbasiskan komponen *Open Source*. Hal ini didasarkan atas pertimbangan fleksibilitas yang ada pada komponen *Open Source* tersebut. Dengan komponen *Open Source* memungkinkan pengembangan dilakukan secara cepat dengan hasil yang baik tanpa menghabiskan dana untuk lisensi. Sebagai komponen utama untuk proses pencarian, sistem ini menggunakan *project Open Source Lucene* sebagai komponen yang membantu proses pengindeksasian dan pencarian dokumen. Sebelum artikel-artikel tersebut diindeks, maka dilakukan terlebih dahulu dilakukan konversi artikel dalam bentuk PDF dan HTML ke dalam bentuk teks yang dilakukan oleh purwa rupa.

Sedangkan untuk melakukan analisis kandungan teks maka digunakan proyek buatan IBM yaitu *Unstructured Information Management Architecture* (UIMA) [4-7]. Pendefinisian data semantik untuk masing-masing artikel dibuat dengan menggunakan XML yang terintegrasi dengan UIMA. UIMA memiliki beberapa komponen utama untuk melakukan analisis kandungan teks dengan menggunakan data semantik yang sudah telebih dahulu didefinisikan seperti *Collection Reader*, *Analysis Engine* dan *CAS Consumer*. *Collection Reader* berfungsi untuk mengumpulkan seluruh file teks yang akan dianalisa dan mengembalikan tipe CAS yang meliputi artikel-artikel yang akan

dianalisa. Kemudian, *Analysis Engine* menggunakan CAS tersebut untuk menganalisa kandungan teks dan menghasilkan *CAS consumer* yang kaya akan *tag*. Selanjutnya, *CAS Consumer* menggunakan CAS tersebut untuk menghasilkan beberapa *tag* untuk masing-masing artikel. *Tag-tag* yang dihasilkan untuk masing-masing artikel akan disimpan di basis data oleh purwa rupa untuk mempercepat proses pencarian.

Tentu saja agar sistem dapat berjalan dengan baik maka dibutuhkan komponen lainnya, antara lain Tomcat dan MySql. Sedangkan sebagai bahasa pemrograman digunakan Java dengan berbagai pustaka yang mendukung. Dari penjelasan diatas, maka dapat disimpulkan tiga proses utama yang dijalankan oleh mesin pencari ini adalah melakukan indeks artikel, analisa kandungan teks dalam suatu kalimat dan pencarian dengan menggunakan Lucene. Gambaran umum sistem arsitektur dari mesin pencari ini dapat dilihat pada Gambar 1.

Ketika pengguna memasukkan kata kunci ke dalam mesin pencari maka mesin pencari akan menampilkan seluruh artikel yang relevan dengan kata kunci yang dimasukkan oleh pengguna. Urutan artikel yang dihasilkan dihitung berdasarkan nilai relevansi dengan:

1. manual tag yang dimasukkan oleh pengguna untuk masing-masing artikel;
2. kemiripan tag dengan kata-kata yang terdapat dalam suatu artikel;
3. banyaknya frekuensi kata kunci yang muncul pada judul suatu artikel; dan
4. banyaknya frekuensi kata kunci yang muncul di konten artikel.

Implementasi proses perhitungan bobot nilai relevansi dari suatu *tag* terhadap kata kunci yang dimasukkan oleh pengguna dapat didefinisikan sebagai fungsi matematika yang dapat dilihat sebagai berikut:

Scoring =

f((p1 x q1)+(p2 x q2)+(p3 x q3)+(p4 x q4)) +g(x)

dimana:

p1 = manual tag
q1 = nilai untuk manual tag
p2 = kemiripan tag
q2 = nilai untuk kemiripan tag
p3 = judul
q3 = nilai untuk judul
p4 = konten
q4 = nilai untuk konten

g(x) = fungsi tambahan yang dapat ditambahkan untuk implementasi di masa depan, contohnya URL, dll

Setelah tahap implementasi, maka purwa rupa mesin pencari ini di-*install* di *server* Departemen Pendidikan Nasional yang berlokasi di Jakarta yang dapat diakses oleh semua orang melalui http://cmsdev.jardiknas.org/TagGenerator/ agar dapat dievaluasi secara *online* oleh para pengguna. Evaluasi dilakukan dengan beberapa ketentuan yaitu:

1. Purwa rupa hanya dapat mengakomodir dengan sebagian kecil anotasi;
2. Fitur yang terdapat di purwa rupa terdiri dari fitur pencarian, kumpulan *tag* yang berelasi dengan kata kunci yang dimasukkan oleh pengguna, penambahan *tag* berdasarkan kata kunci, *paging* hasil pencarian dan fitur "*did you mean"* yang membantu pengguna apabila pengguna salah memasukkan kata kunci;
3. Purwa rupa hanya dapat menggunakan satu simbol *wildcard* di kata kunci yang ingin dicari oleh pengguna; dan
4. Lingkupan data yang telah dianalisa artikelnya hanya mencakup data di *website* Presiden SBY dari tahun 2004 sampai dengan Juni 2007.

Dalam tahap pengujian disajikan evaluasi usabilitas dari proses pencarian serta pemanfaatan *tag* oleh pengguna dalam melakukan pencarian. Evaluasi usabilitas dilakukan melalui dua tahapan. Tahapan evaluasi pertama dengan melibatkan 100 pengguna bertujuan untuk mengidentifikasikan perlakuan pengguna ketika mencari informasi melalui purwa rupa ini. Dan, tahapan evaluasi kedua dengan melibatkan 50 pengguna bertujuan untuk mengetahui perlakuan pengguna dalam menjawab beberapa pertanyaan yang telah disediakan dengan tujuan supaya pengguna mempersempit hasil pencarian mereka.

Skenario evaluasi yang dilakukan oleh pengguna adalah:

1. Pengguna dapat memasukkan beberapa kata kunci yang telah dianotasikan sebelumnya di dalam purwa rupa mesin pencari seperti Presiden, Wapres, Provinsi, Departemen, BUMN, ASEAN, Undang-Undang, SBY, Spanyol, Bali, Beruang, Jubir, dan lain-lain.
2. Setelah pengguna mendapatkan hasil dari purwa rupa maka pengguna akan membandingkan hasil dari mesin pencari lain, dalam hal ini Google, dengan menggunakan kata kunci yang sama. Perbandingan dilakukan dengan memperhatikan:
    a. Apakah implementasi kumpulan *tag* di purwa rupa memberikan keuntungan terhadap pengguna?
    b. Apakah hasil yang ditampilkan pada purwa rupa mesin pencari ini memberikan hasil yang relevan dengan kata kunci yang dimasukkan?
    c. Apakah purwa rupa dapat memberikan

hasil yang lebih baik (misal urutan relevansi artikel, dsb.) dibandingkan mesin pencari lainnya seperti Google?

3. Selanjutnya, pengguna dapat memberikan komentar, saran dan penilaian untuk purwa rupa ini berdasarkan hasil perbandingan yang telah dilakukan pada tahap kedua.

Berdasarkan hasil evaluasi tersebut diaras, maka dapat disimpulkan bahwa purwa rupa mesin pencari dapat memberikan hasil yang relevan dengan kata kunci yang diketikkan oleh pengguna, khususnya untuk kata-kata yang memiliki lebih dari satu arti, contohnya kata ”beruang”, dan fitur kumpulan *tag* yang berelasi dengan kata kunci sangat membantu pengguna dalam pencarian ulang.

5. Kesimpulan dan Saran

Kebutuhan akan mesin pencari yang handal sangat dibutuhkan karena beragamnya informasi yang ada saat ini. Oleh karena itu, dalam penelitian ini dikembangkan purwa rupa mesin pencari yang dapat melakukan analisa kandungan teks dalam suatu kalimat bahasa Indonesia, khususnya untuk mengenali kata yang memiliki lebih dari satu arti. Selain menampilkan artikel-artikel yang terkait dengan kata kunci yang dimasukkan oleh pengguna, purwa rupa ini juga dapat menampilkan kumpulan *tag* (*Cloud of Tags*) yang berkaitan dengan kata kunci yang dimasukkan. *Tag* ini dapat digunakan oleh pengguna untuk pencarian ulang dan mengklasifikasikan informasi umum menjadi informasi yang lebih khusus. Pengembangan purwa rupa ini menggunakan *Open Source Java, Lucene* dan UIMA. Setelah melalui tahap implementasi maka dilakukan evaluasi oleh beberapa pengguna secara *online*. Didemonstrasikan pada purwa rupa yang dioperasikan secara *online*, bahwa penggunaan analisis semantik mempermudah pengguna dalam mencari artikel yang dibutuhkan.

REFERENSI

[1] Allen, James. 1995. *Natural Language Understanding*. The Benjamin/Cummings Publishing Company, Inc.

[2] Uliniansyah, Mohammad Teduh dan Juniar Ganis. 3 Oktober 2007. “Introduction to Bahasa Indonesia and NLP-related Researches in BPP Teknologi.” http://www.tcllab.org/events/uploads/teduh-indonesia.pdf.

[3] Dolan, Bill. 2 Oktober 2007. “Natural Language Processing”. http://research.microsoft.com/nlp/.

[4] Ferrucci dan A. Lally. 2 Oktober 2007. “Building an example application with the Unstructured Information Management Architecture”. http://www.research.ibm.com/journal/sj/433/ferrucci.html.

[5] Ferruci, David A. 2 Oktober 2007. “UIMA and Semantic Search Introductory Overview”. www.research.ibm.com/UIMA/UIMA%20and%20Semantic%20Search%201Q2007.ppt.

[6] IBM. 2 Oktober 2007. “Unstructured Information - The Knowledge Rush”. http://www.research.ibm.com/UIMA/.

[7] IBM. 2 Oktober 2007. “Unstructured Information Management Architecture (UIMA): SDK User's Guide and Reference”. http://dl.alphaworks.ibm.com/technologies/uima/UIMA_SDK_Users_Guide_Reference.pdf.