Perancangan Aplikasi Web Crawler untuk Menghasilkan Dokumen Teks pada Domain Tertentu

Agustino Halim[#1], Rudy Dwi Nyoto[#2], Novi Safriadi[#3]
[#]*Program Studi Teknik Informatika Universitas Tanjungpura*
Jl. Prof. Dr. H. Hadari Nawawi, Pontianak, Kalimantan Barat 78115
[1]agustinohalim@gmail.com, [2]rudydn@gmail.com, [3]bangnops@gmail.com

Abstrak - **Untuk mendapatkan dan menyaring informasi yang dibutuhkan, pengguna Internet dapat menggunakan mesin pencarian (search engine) yang telah tersedia, misalnya Google, Yahoo, Bing, DuckDuckGo dan lain sebagainya. Mesin pencari tersebut melakukan pencarian berdasarkan kata kunci yang dimasukkan oleh pengguna, selanjutnya mencocokkan kata kunci dengan isi konten yang tersebar di Internet. Sehubungan dengan keterbatasan sumber daya komputasi dan waktu, maka dibutuhkan suatu cara untuk mengambil konten yang ada di Internet dalam waktu yang singkat dan dapat diindeks secara otomatis serta tersimpan pada database. Untuk memudahkan pengambilan informasi yang tersebar dan selalu berubah-ubah di Internet dalam jumlah besar diperlukan sebuah web crawler. Fungsi utama Web Crawler adalah melakukan penjelajahan dan pengambilan halaman-halaman web yang ada di Internet. Tujuan penelitian ini adalah menghasilkan aplikasi web crawler untuk menghasilkan dokumen teks pada domain tertentu dalam bidang Teknik Informatika atau komputer dan sejenisnya. Pengujian dilakukan dengan metode Black Box dengan teknik robustness testing, pengujian precision and recall serta pengujian F-Measure. Berdasarkan hasil pengujian, didapatkan nilai Recall sebesar 0,99 dan Precision sebesar 0,61 serta F-Measure sebesar 0,74.**
Kata Kunci : *F-Measure*, Precision, Recall, *Web Crawler, Web Spider*

I. PENDAHULUAN

Pertumbuhan teknologi Internet telah berkembang pesat dan menjadi salah satu kebutuhan sehari-hari. Pengguna Internet di seluruh dunia mempublikasikan sumber daya yang mereka miliki di Internet sehingga memudahkan persebaran dan akses informasi dari mana saja. Untuk mendapatkan dan menyaring informasi yang dibutuhkan, pengguna Internet dapat menggunakan mesin pencarian (search engine) yang telah tersedia, misalnya Google, Yahoo, Bing, DuckDuckGo dan lain sebagainya. Mesin pencari tersebut melakukan pencarian berdasarkan kata kunci yang dimasukkan oleh pengguna, selanjutnya mencocokkan kata kunci dengan isi konten yang tersebar di Internet. Sehubungan dengan keterbatasan sumber daya komputasi dan waktu, maka dibutuhkan suatu cara untuk mengambil konten yang ada di Internet dalam waktu yang singkat dan dapat diindeks secara otomatis serta tersimpan pada database[1].

Untuk memudahkan pengambilan informasi yang tersebar dan selalu berubah-ubah di Internet dalam jumlah besar diperlukan sebuah web crawler. Web Crawler atau dengan kata lain Web Spider ataupun Web Robot merupakan salah satu komponen penting dalam sebuah mesin pencari modern. Fungsi utama Web Crawler adalah melakukan penjelajahan dan pengambilan halaman-halaman web yang ada di Internet. Hasil pengumpulan situs web selanjutnya akan di indeks oleh mesin pencari sehingga mempermudah pencarian informasi di Internet [4].

Perancangan sebuah web crawler yang baik saat ini masih menemui banyak kesulitan. Kesulitan merancang sebuah web crawler dibagi menjadi dua, yaitu kesulitan secara internal dan kesulitan secara eksternal. Secara internal, crawler harus dapat mengatasi besarnya volume data. Sedangkan secara eksternal, crawler harus mengatasi besar dan cepatnya perubahan situs web dan link jaringan yang ada [5].

Data - data yang disimpan merupakan metadata yang ada pada web tersebut, misalnya header, content, footer, dan sebagainya. Dalam implementasi selanjutnya diharapkan data-data yang sudah tersimpan tersebut dapat diakses secara offline untuk berbagai keperluan, salah satunya dapat digunakan sebagai repositori dokumen pembanding pada aplikasi pendeteksian plagiarisme tugas kuliah mahasiswa.

II. URAIAN PENELITIAN

A. *Web Crawler*

Web Crawler adalah program yang menelusuri World Wide Web dengan cara yang metodis, otomatis dan teratur. Istilah lain untuk web crawler adalah ant, automatic indexer, bots, web spiders atau web robots.

Web Crawler adalah salah satu jenis bot atau agen perangkat lunak. Secara umum, proses crawling dimulai dengan list URL yang akan dikunjungi, disebut seeds. Kemudian web crawler akan mengunjungi URL tersebut satu per satu. Setiap page URL yang dikunjungi akan diidentifikasi apakah ada hyperlink di dalamnya. Jika ada maka akan ditambahkan ke dalam list URL yang akan dikunjungi. Ini disebut crawl frontier. URL yang didapat dari crawl frontier ajab dikunjungi secara rekursif dengan beberapa kebijakan tertentu [2].

B. *Uniform Resource Locator*

URL merupakan sistem pengalamatan yang digunakan pada World Wide Web. Di internet URL menggabungkan informasi mengenai jenis protokol yang digunakan, alamat situs dimana resource ditempatkan, lokasi sub directory dan nama file yang digunakan.

Sintak lengkap suatu URL:

Access-methode://server_name1: [port]/directory/file

Contoh:

http://www.microsoft.com/mspress/net/default.asp

Submitted 23-02-2017; Revised 23-02-2017; Accepted 23-02-2017

URL diatas terdiri dari komponen-komponen:

a. http: tipe internet protokol yang digunakan untuk menyimpan dan mengirim informasi.

b. ://: standar pemberian tanda baca URL.

c. www.microsoft.com: nama domain situs dimana resources disimpan.

d. /mspress/net: tempat directory ke resources yang disimpan di komputer yang jauh (dalam hal ini sebuah file).

e. Default.asp: nama file yang dibuka.

URL menyediakan sebuah daftar metode yang konsisten dan mudah dimengerti dari berbagai macam situs internet, terutama pada situs world wide web [3].

III. PERANCANGAN SISTEM

A. *Arsitektur Sistem*

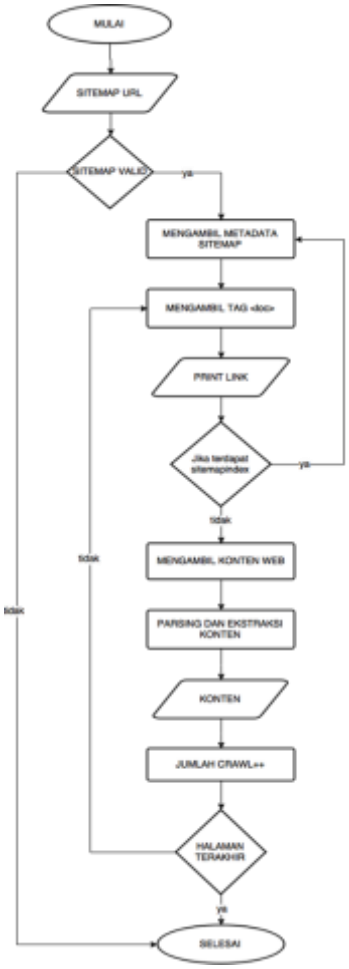
Desain arsitektur sistem akan ditunjukan pada Gambar 1,

Gambar 1. Desain Arsitektur Sistem

Aplikasi web crawler yang dibangun memanfaatkan sitemap untuk mengambil link – link yang ada pada website yang akan dicrawling. Fungsi utama aplikasi ini adalah melakukan pengambilan metadata yang ada pada link dengan masukan berupa sitemap website tersebut. Hasil crawling yang diharapkan berupa dokumen berbentuk plaintext sehingga dapat disimpan pada database untuk digunakan pada repositori dokumen atau pada aplikasi plagiarisme. Secara garis besar, sistem terdiri dari lima komponen utama, yaitu terdiri dari pengguna, PC (personal computer), Internet, website dan crawler.

B. *Flowchart Crawling*

Flowchart Crawling diperlihatkan pada Gambar 2,



Gambar 2. Flowchart Crawling

C. *Pengujian Aplikasi*

Pengujian aplikasi dilakukan dengan menggunakan metode Recall, Precision dan F-Measure. Pengujian Recall dan Precision merupakan pengujian untuk mendapatkan informasi hasil pencarian yang didapatkan oleh Sistem Temu Kembali Informasi (STKI). Hasil pencarian STKI bisa dinilai tingkat *recall* dan *precision* nya. *Precision* dapat dianggap sebagai ukuran ketepatan atau ketelitian, sedangkan *recall* adalah kesempurnaan. Rumus *recall* dapat dilihat sebagai berikut.

$$R = \frac{\text{Jumlah dokumen relevan yang terpanggil}}{\text{Jumlah dokumen relevan yang ada di dalam database}}$$

Dengan R adalah *recall*, maka nilai R didapatkan dengan membandingkan Jumlah dokumen relevan yang terpanggil dengan Jumlah dokumen relevan yang ada di dalam database. Rumus *precision* dapat dilihat sebagai berikut.

$$P = \frac{\text{Jumlah dokumen relevan yang terpanggil}}{\text{Jumlah dokumen yang terpanggil dalam pencarian}}$$

Dengan P adalah Precision. maka nilai P didapatkan dengan membandingkan Jumlah dokumen relevan yang terpanggil dengan jumlah dokumen yang terpanggil dalam pencarian.

Korespondensi : Agustino Halim

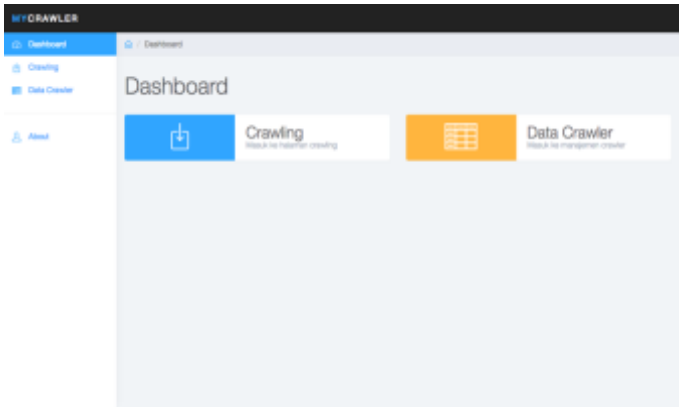
F-Measure merupakan pengukuran yang mengkombinasikan precision dan recall yang diterapjan kedalam deret harmonik. Rumus F-Measure dapat dilihat sebagai berikut.

$$F = 2 \times \frac{\text{precision x recall}}{\text{precison} + \text{recall}}$$

Dengan F adalah F-Measure, maka nilai F didapatkan dengan membandingkan precision kali recall dengan precision ditambah recall kemudian dikali dua.

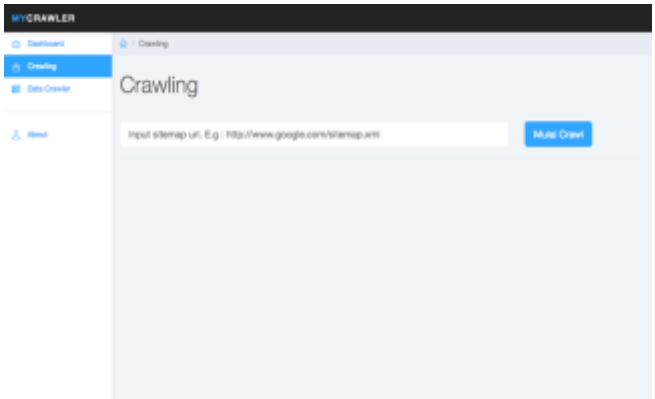
D. Hasil Aplikasi

Aplikasi yang dibangun merupakan aplikasi web crawler untuk menghasilkan dokumen teks pada domain tertentu. Aplikasi ini mengambil link yang ada pada sitemap dan mengambil metadata yang terdapat pada link. Berikut beberapa tampilan hasil perancangan aplikasi, yang diperlihatkan pada Gambar 3 sampai dengan Gambar 5.



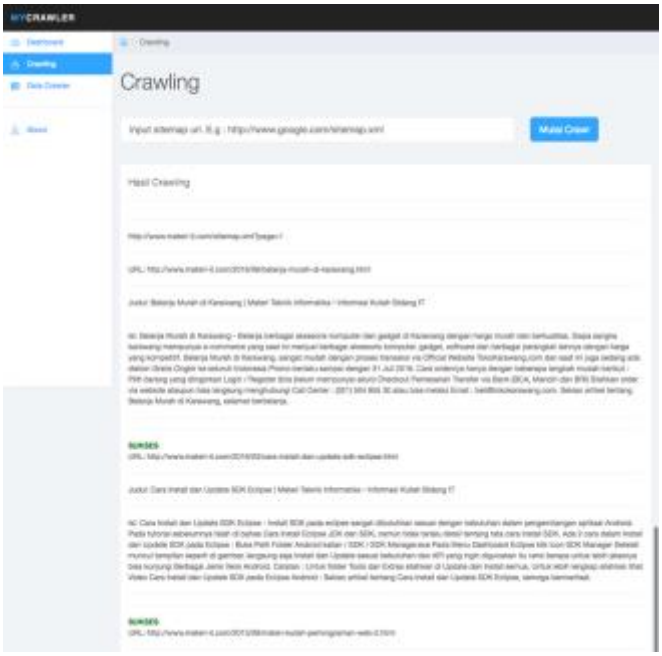
Gambar 3. Tampilan Halaman *Dashboard*

Gambar 3 merupakan tampilan dari menu utama.



Gambar 4. Tampilan Halaman

Gambar 4 merupakan tampilan halaman hasil crawling.



Gambar 5. Tampilan Hasil Crawling

Gambar 5 merupakan tampilan halaman hasil dari crawler

E. Hasil Pengujian

1. Pengujian Recall dan Precision

Tujuan pengujian recall dan precision adalah untuk mendapatkan informasi hasil pencarian yang didapat oleh Sistem Temu Kembali Informasi (STKI). Pada pengujian proses ini digunakan data website yang telah disediakan sebanyak 30 website blogspot yang mengandung materi di bidang teknik informatika.

Pada Tabel 1, memperlihatkan pengujian *recall* dan *precision*

No.	Nama Website	Jumlah				
		Halaman Terpindai	Terambil	Tidak Terambil	Relevan	Tidak Relevan
1	http://126599035-si.blogspot.co.id/sitemap.xml	20	20	0	19	1
2	https://berlati.blogspot.co.id/sitemap.xml	254	253	1	191	59
3	http://algoritma-program.blogspot.co.id/sitemap.xml	143	143	0	143	0
4	http://daniawehbouse.blogspot.co.id/sitemap.xml	35	35	0	33	2
5	http://www.materi-it.com/sitemap.xml	489	488	1	488	0
6	http://www.kajianpustaka.com/sitemap.xml	454	454	0	90	364
7	http://choosetozone.blogspot.co.id/sitemap.xml	115	115	0	34	81
8	http://www.gatewan.com/sitemap.xml	386	386	0	386	6
9	http://tasiBagus.blogspot.co.id/sitemap.xml	165	165	0	60	105
10	http://blog-algoritma-programmer.blogspot.co.id/sitemap.xml	129	129	0	129	0
11	http://www.pengartiansih.com/sitemap.xml	730	730	0	221	09
12	http://www.unix-informatika.com/sitemap.xml	272	272	0	124	148
13	http://penalitianteknik.blogspot.co.id/sitemap.xml	30	30	0	30	0
14	http://www.indrasatya.com/sitemap.xml	35	35	0	27	8
15	http://pendidikan-informatika.blogspot.co.id/sitemap.xml	30	30	0	30	0
16	http://www.analtkj.net/sitemap.xml	103	103	0	102	1
17	http://cerna-informatika.blogspot.co.id/sitemap.xml	44	44	0	26	18
18	http://komponen-dasar-blog.blogspot.co.id/sitemap.xml	613	612	1	134	478
19	http://ilmu-komputer-ind.blogspot.co.id/sitemap.xml	50	50	0	50	0
20	http://teknikinformatika-ect.blogspot.co.id/sitemap.xml	168	168	0	168	0
21	http://blogkuliah-ti.blogspot.co.id/sitemap.xml	8	8	0	8	0
22	https://gurulnformatika.blogspot.co.id/sitemap.xml	247	247	0	194	53
23	http://irwandp.blogspot.co.id/sitemap.xml	59	59	0	59	0
24	http://codeoke.blogspot.co.id/sitemap.xml	9	9	0	9	0
25	http://kumpulanmateripeering.blogspot.co.id/sitemap.xml	9	9	0	5	4
26	https://gratabang.blogspot.co.id/sitemap.xml	185	185	0	130	55
27	http://uchmad-aseen.blogspot.co.id/sitemap.xml	31	31	0	31	0
28	http://technologies-it.blogspot.co.id/sitemap.xml	5	5	0	5	0
29	http://www.bloginformatika.com/sitemap.xml	77	77	0	46	31
30	http://herlandasrl.blogspot.co.id/sitemap.xml	90	90	0	65	25
	TOTAL	4985	4982	3	3037	1448

Tabel 1
Tabel Hasil Pengujian *Recall* dan *Precision*

Berdasarkan hasil pengujian crawling pada Tabel 1, aplikasi *web crawler* yang dibangun berhasil melakukan *crawling* halaman web sebanyak 4982 halaman dengan jumlah halaman relevan sebanyak 3037 halaman dan tidak relevan

Korespondensi : Agustino Halim

sebanyak 1448 halaman dari 30 website blogspot terpilih. Berikut adalah hasil perhitungan pengujian *recall* dan *precision*.

Hasil Pengujian Recall

$$Recall = \frac{Jumlah\ dokumen\ relevan\ yang\ terambil}{Jumlah\ dokumen\ yang\ relevan\ yang\ ada\ dalam\ database} = \frac{3037}{3037+3} = 0.99$$

Hasil Pengujian Precision

$$Precision = \frac{Jumlah\ dokumen\ relevan\ yang\ terpanggil}{Jumlah\ dokumen\ yang\ terpanggil\ dalam\ pencarian} = \frac{3037}{3037+1448} = 0.61$$

Hasil Pengujian F-Measure

$$FMeasure = 2 \times \frac{precision \times recall}{precision + recall} = 2 \times \frac{0.61 \times 0.99}{0.61 + 0.99} = 0.74$$

Dari hasil perhitungan kita dapatkan nilai *recall* sebesar 0.99, nilai *precision* sebesar 0.61 dan nilai *f-measure* sebesar 0.74. Hal ini menunjukkan bahwa aplikasi dapat mengambil url dari sitemap dan metadata yang diharapkan dengan cukup baik. Berbeda dengan dan sistem belum dapat menghilangkan link yang tidak terdapat artikel didalamnya.

F. Analisis Hasil Pengujian

Rincian hasil analisis pengujian aplikasi crawling yang telah dilakukan adalah sebagai berikut:

1. Berdasarkan hasil pengujian black box menggunakan metode *robustness testing*, aplikasi *web crawler* yang dibangun belum dapat menghilangkan konten iklan dari google yang terdapat pada artikel disebabkan iklan tersebut disisipkan pada awal dan akhir artikel.
2. Berdasarkan hasil pengujian recall dan precision diperoleh nilai recall yang relatif besar disebabkan oleh aplikasi web crawler yang dibuat dapat mengambil metadata dengan baik.
3. Berdasarkan hasil pengujian recall dan precision diperoleh hasil precision yang relatif kecil, dikarenakan website yang diambil kontennya bercampur dengan materi – materi lainnya, tidak hanya materi di bidang Teknik Informatika.

IV. KESIMPULAN/RINGKASAN

Berdasarkan hasil implementasi dan hasil analisis pengujian terhadap aplikasi *web crawler* dapat disimpulkan bahwa aplikasi *web crawler* dapat mengambil *link* yang terdapat pada *sitemap* dan mengambil isi metadata yang terdapat pada *link* tersebut. Dan berdasarkan hasil pengujian black box dengan metode *robustness testing*, aplikasi belum dapat menghilangkan konten iklan dari google yang terdapat pada artikel. Sedangkan berdasarkan hasil pengujian *recall* dan *precision*, diperoleh nilai recall sebesar 0.99 dan precision sebesar 0.61 serta F-Measure sebesar 0.74. hasil precision yang relatif kecil, dikarenakan website yang diambil kontennya bercampur dengan materi – materi lainnya, tidak hanya materi di bidang Teknik Informatika.

DAFTAR PUSTAKA

[1] Sulastri, dan E. Zuliarso, 2010. "Aplikasi Web crawler Berdasarkan Breadth First Search dan Back-Link". Jurnal Teknologi Informasi DINAMIK Vol. XV, No.1, Hal 52-56

[2] Shkapenyuk, Vladislav dan Suel Torsten. 2002. "Design and Implementation of a High-Performance Distributed Web Crawler". Brooklyn, New York: CIS Department, Polytechnic University

[3] Mardiana, Tari. 2012. "Mesin Pengindikasi Kemiripan untuk Dokumen Berbahasa Indonesia". Jogyakarta, Indonesia: Universitas Gadjah Mada.

[4] Zuliarso, Eri dan K. Mustofa. 2009. "Crawling Web berdasarkan Ontology". Jurnal Teknologi Informasi DINAMIK Vol. XIV, No. 2, Hal. 105-112.

[5] Cho, Junghoo., Hector Garcia-Molina, dan Lawrence Page. 1998. "*Efficient crawling through URL ordering*". In Proceedings of the Seventh Internasional World Wide Web Conference. Hal. 161-172.

Korespondensi : Agustino Halim