SKRIPSI

INCREMENTAL WEB CRAWLING PADA LINGKUNGAN HADOOP

Melinda Nur Abianti

NPM: 2014730012

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2018

UNDERGRADUATE THESIS

INCREMENTAL WEB CRAWLING ON HADOOP ENVIRONMENT

Melinda Nur Abianti

NPM: 2014730012

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2018

LEMBAR PENGESAHAN

INCREMENTAL WEB CRAWLING PADA LINGKUNGAN HADOOP

Melinda Nur Abianti

NPM: 2014730012

Bandung, 31 Mei 2018

Menyetujui,

Pembimbing

Gede Karya, M.T., CISA, IPM

Ketua Tim Penguji

Chandra Wijaya, M.T.

Anggota Tim Penguji

Rosa De Lima, M.Kom.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

***INCREMENTAL WEB CRAWLING* PADA LINGKUNGAN HADOOP**

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 31 Mei 2018

Melinda Nur Abianti
NPM: 2014730012

ABSTRAK

Web crawler merupakan perangkat lunak yang melakukan *web scanning* dan pengindeksan *Uniform Resource Locator* (URL) secara periodis. Salah satu permasalahan dari *web crawler* tradisional adalah ketidak efektifan proses *crawling* dikarenakan pengubahan konten *web* setiap periode tidak diperhitungkan untuk menentukan periode waktu *web scanning* selanjutnya. Pada penelitian ini dibangun *web crawler* yang menerapkan salah satu teknik *web crawling*, yaitu *incremental web crawling* untuk menyelesaikan permasalahan tersebut.

Perangkat *incremental web crawling* dibangun berdasarkan rancangan dari A.K. Sharma dan Ashutosh Dixit yang menerapkan algoritma *Self Adjusting Refresh Time Calculator Module* (SARTCM). Algoritma tersebut digunakan untuk menentukan periode waktu *web scanning* URL selanjutnya berdasarkan probabilitas pengubahan konten *web* URL setiap periode *crawling*. Informasi pengubahan konten *web* dan pengubahan periode waktu *crawling* pada setiap versi sebuah URL disimpan pada basis data NoSQL HBase yang memiliki fitur *versioning* untuk penyimpanan sebuah data dengan banyak versi. Untuk meningkatan kecepatan akses data, basis data HBase dibangun di atas Hadoop untuk melakukan penyimpanan terdistribusi.

Terdapat dua perangkat lunak yang dibangun pada penelitian ini, yaitu perangkat lunak situs induk dan agen *crawler*. Situs induk dibangun menggunakan *platform Java 2 Platform, Enterprise Edition* (J2EE) yang memiliki fitur untuk memasukkan URL untuk di-*crawl*, memasukkan informasi *crawler*, mencari konten berdasarkan URL yang di-*crawl*, melihat *log* pencarian dan proses *crawling*, melihat status URL. Agen *crawler* digunakan untuk melakukan proses *incremental web crawling*. Pengujian fungsional dilakukan untuk menguji fitur pada situs induk, dapat disimpulkan bahwa situs induk berhasil dibangun dan seluruh fungsi berjalan dengan baik. Pengujian performa dilakukan untuk menguji tingkat *scalability* pada HBase yang diterapkan pada agen *crawler* dan situs induk, dapat disimpulkan dari salah satu skenario pengujian bahwa semakin banyak *region server* digunakan, maka baris data URL yang dihasilkan akan semakin besar.

Kata-kata kunci: *web crawler*, *incremental web crawling*, Hadoop, NoSQL HBase, J2EE

ABSTRACT

Web crawler is a software which performs web scanning and *Uniform Resource Locator* (URL) indexing periodically. One of the problems regarding traditional web crawlers is the ineffectiveness of the crawling process due to the change of web content each period is not taken into account to determine the next web scanning time period. To resolve the problem, one of web crawling techniques, incremental web crawling, is being built in this research.

The incremental web crawling software is built based on the design of A.K. Sharma and Ashutosh Dixit that implements *Self Adjusted Refresh Time Calculator Module* (SARTCM) algorithm. The algorithm is used to determine the time period of the next URLâĂŹs web scanning based on the probability of the URLâĂŹs web content changes of each crawling period. All version of web contentâĂŹs information and crawling time period of an URL are stored in the HBase NoSQL database that has versioning features for storing a data with multiple versions. To increase the speed of data access, the HBase database is built on top of Hadoop for distributed storage.

There are two software that is built on this research, parent site and agent crawler. Parent site is built on top of *Java 2 Platform, Enterprise Edition* (J2EE) platform that has features to enter URLs for crawling, enter crawler information, search contents based on crawled URLs, views search and crawling processes logs, and views URL's status. Crawler agent is used for performing incremental web crawling processes. Functional testing is used for determine parent site feature, it can be concluded that parent site software runs well. Perform testing is used for testing HBase's scalability feature that is implemented on crawler agent and parent site, based on the result of one of the performance testing scenario, it can be concluded that the more region server used, the number of rows of data URLs generated will be bigger.

Keywords: *web crawler*, *incremental web crawling*, Hadoop, NoSQL HBase, J2EE

Dipersembahkan untuk Tuhan, Orang tua, Dosen Pembimbing,
Kekasih, dan Teman-Teman

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa karena atas berkat rahmatNya penulis dapat menyelesaikan tugas akhir ini yang berjudul "Incremental Web Crawling pada Lingkungan Hadoop". Adapun tugas akhir ini disusun untuk memenuhi salah satu persyaratan untuk menyelesaikan pendidikan di Fakultas Teknologi Informasi dan Sains pada Program Studi Teknik Informatika di Universitas Katolik Parahyangan Bandung. Dalam menyusun skripsi ini penulis menerima banyak sekali bantuan baik secara jasmani dan rohani dari berbagai pihak yang terlibat baik secara langsung maupun tidak langsung. Oleh karena itu, melalui kesempatan ini penulis hendak mengucapkan terima kasih yang sebesar-besarnya kepada:

- Allah SWT, asal segala kelancaran dan kemudahan yang diberikan-Nya sehingga penulis dapat menyelesaikan kuliahnya tanpa kendala.
- Orang tua penulis yaitu Wiwit Eko Khristianto dan Eva Dianita, serta kakak penulis yaitu Maudy Nur Avianti yang telah mengajarkan kemandirian pada penulis dan mendoakan dan mendukung penulis selama penyusunan skripsi.
- Bapak Gede Karya, M.T., CISA, IPM selaku dosen pembimbing yang telah memberi dorongan mental agar penulis tetap teguh mengerjakan skripsi dan memberi masukan selama penyusunan skripsi.
- Bapak Chandra Wijaya, M.T. selaku dosen penguji utama yang telah memberikan kritik dan saran untuk skripsi, dan Ibu Rosa De Lima, M.Kom. selaku penguji pendamping yang juga telah memberikan kritik dan saran untuk skripsi dan juga rela mencurahkan waktu dan mendukung penulis saat penulis mengalami kegalauan saat menjalankan skripsi.
- Staf Tata Usaha Fakultas Teknologi Informasi dan juga Staf Pekarya yang telah melayani penulis dalam hal administrasi perkuliahan maupun tempat penulis mencurahkan keluh kesah.
- Kekasih penulis yaitu Reanta Indra Putra Pratama, yang telah menemani penulis dalam suka dan duka, teman untuk bertukar pikiran, teman yang memberi kritik, saran, dan dukungan, serta memotivasi penulis untuk segera menyelesaikan skripsi.
- Lina Afriyani, teman terdekat penulis yang selalu menyemangati penulis selama suka dan duka dan memberikan dukungan untuk terus berusaha dalam menyelesaikan tugas akhir dan juga tempat berkeluh kesah.
- Faza Hunafa, teman terdekat penulis yang juga selalu menyemangati penulis selama suka dan duka dan memberikan dukungan untuk terus berusaha dalam menyelesaikan tugas akhir dan juga tempat berkeluh kesah.
- Rekan-rekan di Informatika Unpar, yaitu Agina Rinda, Reza Reynaldi Hasan Haznam, Sapta Hadi Kesuma, Jovanka Helen, Kevin Pratama, Stephanie Tania, Prayogo Cendra, Daud Andrew Gorgha, dan lain-lain; yang telah mendukung untuk menyelesaikan tugas akhir ini hingga penulis lulus.

- Pihak-pihak lain yang belum disebutkan, yang telah memberikan bantuan dalam penyusunan skripsi.

  Akhir kata, penulis menyadari bahwa skripsi ini tidak lepas dari kekurangan. Namun penulis berharap skripsi ini dapat memberikan kontribusi baik untuk penelitian atau pembelajaran selanjutnya.

Bandung, Mei 2018

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Web crawler merupakan perangkat lunak yang melakukan *web scanning* dan membuat indeks dari *Uniform Resource Locator* (URL) yang dicari dengan metode tertentu. *Web crawler* dapat digunakan di berbagai tempat, contohnya di dalam *search engine* untuk mengumpulkan informasi yang berguna saat mencari data, sebagai perangkat lunak untuk menyisir *web* untuk mencari kata/kumpulan kata tertentu yang digunakan, dan melihat *trend* pasar atas dasar pencarian kata tertentu.

Dalam melakukan *crawling*, *web crawler* memiliki metode-metode yang dapat digunakan, yang masing-masing memiliki kelebihan tersendiri. Contohnya adalah *distributed crawling*, *incremental crawling*, *focused crawling*, *deep crawling*, dan lainnya. Metode *incremental crawling* merupakan metode *crawling* yang secara *increment* memperbaharui konten *web* dengan durasi waktu yang tidak tetap, tergantung dari seberapa banyak pengubahan konten *web* yang telah terjadi sebelumnya. Jika dibandingkan dengan metode *crawling* tradisional yang secara periodik mengganti dokumen lama dengan dokumen baru, penggunaan metode *incremental crawling* berguna untuk mengurangi nilai probabilitas konten yang sama dibandingkan konten terdahulunya, yang mengakibatkan proses *crawling* dianggap tidak terlalu efektif karena mengunduh informasi konten yang sama.

Pada penelitian ini, dibuat perangkat lunak yang menerapkan metode *incremental crawling* untuk menangani salah satu permasalahan *crawling*, yaitu pengubahan konten *web* seiring waktu. Pengubahan konten dalam suatu *web* belum tentu dilakukan secara periodik dan pada saat *crawling* dilakukan, konten tidak dapat dipastikan pengubahannya. Hal ini mengakibatkan metode *crawling* tradisional menjadi tidak efisien. Maka dari itu, penelitian ini menggunakan *incremental crawling* sebagai acuan metode *crawling* untuk menangani permasalahan yang telah dijelaskan di atas.

Algoritma *incremental crawling* pada penelitian ini diimplementasikan berdasarkan rancangan A.K. Sharma dan Ashutosh Dixit. Pada rancangan tersebut, diimplementasikan algoritma *Self Adjusting Refresh Time Calculator Module* (SARTCM). Algoritma tersebut berfungsi untuk menentukan frekuensi *refresh time* sebuah URL pada setiap periode proses *crawling* berdasarkan probabilitas pengubahan elemen (p_c), batas bawah probabilitas pengubahan elemen (p_l), dan batas atas probabilitas pengubahan elemen (p_g) yang didapatkan dari akumulasi seluruh periode frekuensi pengubahan elemen konten web URL tersebut. Probabilitas pengubahan elemen pada sebuah konten web dapat ditentukan dengan mengamati pengubahan konten web saat ini dengan konten web pada periode-periode *crawling* sebelumnya, juga dapat ditentukan dengan mengamati elemen pada *Hypertext Transfer Protocol* (HTTP) Response URL, misalnya adalah elemen ETag dan Last-Modified.

Perangkat lunak dibuat dengan bantuan *framework* Hadoop. Hadoop adalah *framework open source* berbasis Java di bawah lisensi Apache untuk men-*support* aplikasi yang berjalan pada *big data*. Hadoop menyediakan beragam ekosistem yang memiliki beragam kegunaan berbeda. Di antara beragam ekosistem tersebut, terdapat ekosistem yang menyediakan mekanisme *versioning* yang menyimpan versi pembaharuan dari suatu nilai yang dimasukkan ke dalam suatu sel dalam basis data Hadoop, yang berguna dalam menyelesaikan permasalahan pengubahan konten seiring waktu. Dengan adanya *versioning*, dapat diketahui kapan saja dan pengubahan apa saja yang terjadi

dalam suatu konten *web*, sehingga membantu pembuatan perangkat lunak yang menerapkan metode *incremental crawling*. Selain itu, HBase memiliki sifat *scalable* yang dapat dimanfaatkan untuk mengolah data berukuran besar, salah satunya adalah mengolah informasi URL pada perangkat lunak yang dibangun.

Pada penelitian ini, dilakukan pengimplementasian *incremental crawling* pada perangkat lunak. Di dalam perangkat lunak tersebut, pengguna dapat masuk sebagai admin untuk memasukkan URL untuk di-*crawl*, melihat hasil proses *incremental crawling* dari URL yang dimasukkan, dan memasukkan informasi *crawler*. Selain itu, pengguna juga dapat mencari konten yang terkandung pada URL yang di-*crawl* tanpa perlu melakukan akses sebagai admin.

1.2 Rumusan Masalah

Berikut ini adalah rumusan masalah dari penelitian ini:

1. Bagaimana mekanisme *incremental crawling* untuk menyelesaikan permasalahan pengubahan konten *web* seiring waktu?
2. Bagaimana mekanisme penyimpanan informasi *versioning* pada ekosistem Hadoop untuk menyelesaikan permasalahan pengubahan konten *web* seiring waktu?

1.3 Tujuan

Berdasarkan rumusan masalah yang telah diuraikan sebelumnya, maka tujuan dari penelitian ini adalah:

1. Memahami mekanisme dan hasil dari *incremental crawling* untuk menyelesaikan permasalahan pengubahan konten *web* seiring waktu.
2. Memahami mekanisme penyimpanan informasi *versioning* pada ekosistem Hadoop untuk menyelesaikan permasalahan pengubahan konten *web* seiring waktu.

1.4 Batasan Masalah

Rumusan masalah yang telah disebutkan di atas masih memiliki ruang lingkup yang cukup luas. Karena keterbatasan waktu dan kemampuan yang dimiliki, maka penelitian ini hanya memfokuskan pada batasan masalah sebagai berikut:

1. Informasi yang diperoleh pada hasil *crawling* bersifat tekstual. Oleh karena itu, konten hasil *crawling* yang dapat dicari bersifat tekstual.
2. Informasi data diri (nama, *username*, dan *password*) dari admin perangkat lunak tidak dapat diperbaharui untuk menyederhanakan fungsionalitas perangkat lunak.
3. Satuan *refresh time* pada proses *crawling* memiliki satuan menit.
4. Penghitungan p_l dan p_g menggunakan metode penghitungan kuartil Mendenhall and Sincich.
5. Nilai HTTP *Response* yang digunakan untuk memeriksa pengubahan elemen adalah ETag dan Last-Modified.
6. Konten *Hypertext Markup Language* (HTML) yang disimpan pada basis data merupakan seluruh elemen yang berada pada *tag body* HTML.
7. Fokus pengerjaan pada penelitian ini adalah melihat pengubahan *refresh time* setiap versi URL berdasarkan pengubahan elemen yang terjadi pada sebuah URL.

1.5 Metodologi

Berikut ini adalah langkah-langkah yang dilakukan dalam membangun perangkat lunak:

1. Studi pustaka mengenai konsep *web crawler*.
2. Studi pustaka dan eksplorasi mengenai sistem terdistribusi berbasis Hadoop.
3. Studi pustaka mengenai *Java 2 Platform, Enterprise Edition* (J2EE)
4. Analisis pemilihan *database* pada ekosistem Hadoop.
5. Studi teknik mengenai *incremental crawling*.
6. Studi penyimpanan informasi *versioning* dan penyimpanannya pada ekosistem Hadoop.
7. Merancang dan mengembangkan sistem perangkat lunak *web crawler*, yang terdiri atas situs induk dan agen *crawler*.
8. Menguji sistem perangkat lunak *web crawler* baik fungsional maupun *performance*-nya.
9. Menulis dokumen penelitian.

1.6 Sistematika Pembahasan

Sistematika penulisan dalam penelitian ini adalah sebagai berikut:

- Bab 1 Pendahuluan
  Bab ini membahas tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, serta sistematika pembahasan.
- Bab 2 Dasar Teori
  Bab ini membahas tentang teori-teori dasar yang menunjang penelitian yang dilakukan yaitu teori *web crawler*, struktur data *queue*, *incremental web crawler*, Hadoop, *Not only* SQL (NoSQL) HBase, J2EE, format komunikasi HTTP, dan Jsoup.
- Bab 3 Analisis
  Bab ini membahas tentang analisis mengenai teknik *incremental crawling* yang digunakan untuk melakukan *incremental crawling*, analisis perangkat lunak, serta hasil eksplorasi mengenai Hadoop dan lingkungannya yang terdiri atas Hadoop, Zookeeper, dan HBase.
- Bab 4 Perancangan
  Bab ini membahas tentang perancangan yang dibutuhkan untuk membangun perangkat lunak, yaitu perancangan antarmuka, perancangan basis data, diagram kelas rinci dan fungsi di dalamnya.
- Bab 5 Implementasi dan Pengujian
  Bab ini membahas tentang lingkungan implementasi, implementasi perangkat lunak (baik secara antarmuka, basis data, maupun fungsi), pengujian fungsional, pengujian *performance*, dan kesimpulan hasil pengujian.
- Bab 6 Kesimpulan dan Saran
  Bab ini membahas tentang kesimpulan dari penelitian yang dilakukan dan saran untuk pengembangan penelitian selanjutnya.