ANALISIS DAN IMPLEMENTASI EKSTRAKSI WEB CONTENT DENGAN METODE AUTOMATIC DATA EXTRACTION MENGGUNAKAN ALGORITMA INSTANCE-BASED DATA EXTRACTION

ANALYSIS AND IMPLEMENTATION OF WEB CONTENT EXTRACTION WITH AUTOMATIC DATA EXTRACTION METHOD USING INSTANC

Achmad Faiz Adrianto[1], Kiki Maulana[2], Arie Ardiyanti Suryani[3]

[1]Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

---

Abstrak

Sebagian besar metode ekstraksi data pada halaman web menggunakan wrapper induction dan automatic data extraction. Metode automatic data extraction muncul karena metode sebelumnya dianggap terlalu rumit. Dalam proses ekstraksi data, automatic data extraction membentuk pattern yang akan dicocokan dengan tag HTML pada halaman web.

Pada Tugas Akhir ini mengimplementasikan metode automatic data extraction dengan menggunakan algoritma yang disebut IDE (Instance-based Data Extraction). Teknik ini melibatkan user dalam pembentukan pattern dengan memberikan label pada halaman web. Pada proses instance-based data extraction ini ada tiga langkah yang utama yaitu, page labeling, similarity measure dan data extraction.

Ketepatan dalam membentuk pattern ekstraksi dapat dilakukan dengan cara mengisi nilai range node sebanyak jumlah node yang terdapat dalam satu template dari target item.

Performansi algoritma IDE dipengaruhi oleh nilai range node yang diberikan. . Jika node yang diambil semakin mendekati pattern target item maka performansi akan semakin baik. Selain itu jenis website yang diekstrak juga ikut mempengaruhi performansi. Website yang memiliki pattern target item (struktur HTML dari data yang akan diekstrak) sederhana akan lebih mudah untuk diekstrak.

Ketika data yang akan diekstrak tidak memiliki pattern yang unik maka algoritma IDE akan kesulitan untuk mengekstrak data yang sesuai dengan keinginan user. Akibatnya di dalam hasil ekstraksi masih terdapat data yang tidak relevan.

Tahap analisis dan pengujian dengan parameter pengujian berupa recall ratio dan precision ratio memberikan hasil bahwa algoritma IDE yang dibangun terbukti bisa mendapatkan informasi sesuai dengan keinginan user meskipun ada beberapa noise.

Kata Kunci : Automatic Data Extraction, pattern, pattern target item, page labeling, similarity measure, data extraction

---

Abstract

Most of the data extraction method on web pages using the wrapper induction and automatic data extraction. Automatic data extraction method occurs because the previous method was considered too complicated. In the process of data extraction, automatic data extraction form a pattern that will be fitted with the HTML tags on web pages.

This Final project implemented automatic data extraction method with use algorithm is called IDE (Instance-based Data Extraction). This technique involves the user in forming pattern by labels the web page. In the process of instance-based data extraction are three main steps, i.e, page labeling, similarity measure and data extraction.

Accuracy in shaping the pattern extraction can be done by filling the range node as much as nodes contained in a template of the target item.

IDE algorithm performance is affected by the given value of range node. If the node is drawn closer and closer to the pattern target items then his performance will improve. Moreover, type of website that is extracted also affect performance. Website which has simple pattern target item (the HTML structure of the data to be extracted) is easier to be extracted.

When the extracted data does not have a unique pattern, the IDE algorithm's will be difficult to extract data according to user desires. There are irrelevant data as a result of the extraction results.

Phase implementation and analysis with testing parameters such as recall ratio and precision ratio shown that built IDE algorithm is proved to obtain information in accordance with the user desires though it is noise.

Keywords : Automatic Data Extraction, pattern, pattern target item, page labeling, similarity measure, data extraction

1. Pendahuluan

1.1. Latar belakang

Seiring dengan pesatnya pertumbuhan sumber informasi yang tersedia di *World Wide Web* pada jaringan internet, bermunculan juga teknik-teknik *data mining* yang ditujukan untuk mempermudah pemakai dalam memperoleh data yang ada sesuai dengan keinginannya. Akan tetapi terdapat permasalahan untuk mengidentifikasi potongan informasi relevan pada halaman web karena sering dikacaukan dengan isi tidak relevan seperti iklan, *navigation-panel*, *copyright notice* dan lain-lain disekeliling inti isi dari halaman web. Diperlukan suatu alat untuk mengekstrak informasi dari halaman web untuk menyediakan nilai tambah layanan, yaitu mengekstrak data(*instance*) dalam daerah khusus(*template*) pada halaman web yang sesuai dengan keinginan user.

Ada beberapa teknik yang sudah ada untuk mengekstrak data dari halaman web termasuk teknik manual dan teknik otomatis. Pendekatan manual untuk mengeluarkan data dari sumber-sumber web adalah dengan cara menulis program-program khusus, yang disebut *wrappers*, bahwa data mengidentifikasi hal penting dan memetakan mereka untuk beberapa format yang sesuai. Mengembangkan *wrappers* secara manual memiliki banyak kelemahan, terutama karena kesulitan dalam penulisan dan pemeliharaan *wrappers*.

Teknik otomatis atau sering disebut dengan *Automatic Data Extraction* merupakan cara lain untuk mengekstrak data. Omini [1], Information Extraction Based on Pattern Discovery(IEPAD) [4] dan Mining Data Record(MDR) merupakan contoh algoritma dari metode *Automatic Data Extraction*. Ketiga algoritma tersebut membentuk *pattern* untuk ekstraksi data. Akan tetapi *pattern* yang dihasilkan tidak spesifik karena semuanya ditentukan oleh sistem. Oleh karena itu ekstraksi data yang dihasilkan belum tentu sesuai dengan keinginan user.

Instace-Based Dara Extraction (IDE) merupakan salah satu dari sekian banyak algoritma pada metode *Automatic Data Extraction*. Algoritma ini masuk dalam kategori *Automatic Data Extraction* karena mencari *pattern* yang berulang kemudian menggunakan *pattern* tersebut untuk mengkestrak data. Diharapkan dengan adanya keterlibatan user dalam pemilihan data yang akan diekstrak memberikan pengaruh yang signifikan terhadap hasil ekstraksi.

Dalam penyusunan Tugas Akhir ini akan dilakukan implementasi metode ADE menggunakan algoritma IDE. Dalam proses ekstraksi data, metode ini melibatkan user dalam memberikan label pada halaman web sehingga dapat memberikan hasil yang sesuai dengan keinginan user. Algoritma IDE hanya membutuhkan satu halaman web untuk membuat *pattern* yang akan digunakan untuk mengkestrak data.

1.2. Perumusan Masalah

Berdasarkan latar belakang tersebut, maka permasalahan yang akan dibahas adalah sebagai berikut :

1. Bagaimana mengimplementasikan metode *Automatic Data Extraction* dengan algoritma IDE untuk ekstraksi data pada halaman web.

2. Bagaimana menentukan *pattern* yang dapat secara tepat mengekstraksi data sehingga sesuai dengan keinginan user.
3. Apa saja faktor yang mempengaruhi performansi algoritma IDE.
4. Bagaimana melakukan pengukuran dan pengujian untuk mengetahui performansi dari sistem IDE yang dibangun.

Batasan masalah untuk penelitian ini adalah :

1. Dokumen web yang digunakkan pada penelitian ini terbatas pada dokumen HTML.
2. Kategori dokumen web terbatas pada website jenis daftar produk dan jenis berita.
3. Implementasi yang dibuat dalam tugas akhir ini menekankan hanya pada proses mendapatkan inti informasi dari sebuah halaman web yang berupa teks dan image, jadi tidak termasuk link ke halaman lain.
4. Item yang diekstrak terbatas pada pilihan yang disediakan oleh aplikasi.
5. Yang diamati hanyalah performansi yang berupa *recall ratio* dan *precission ratio*.

1.3. Tujuan

Tujuan dari dilakukannya penelitian ini adalah:

1. Mengetahui ketepatan *pattern* yang di bentuk dengan menggunakan metode *Automatic Data Extraction* dengan algoritma IDE.
2. Menghasilkan ekstraksi data sesuai keinginan user.
3. Menganalisa faktor-faktor yang mempengaruhi performansi berupa *range node* dan jenis website inputan.
4. Menganalisa sistem yang telah dibangun ini dilihat dari *recall ratio* dan *precision ratio*.

1.4. Metodologi penyelesaian masalah

Penelitian ini akan mengimplementasikan ekstraksi *Web Content* dengan *Automatic Data Extraction* menggunakan metode *Instance-based Data Extraction.*

1. Studi Literatur :

Pencarian referensi dan sumber-sumber lain yang dapat digunakan sebagai acuan dalam penelitian dan pembelajaran ekstraksi pada *Web Content*. Meliputi tentang proses pemberian label pada web, ektraksi struktur data, dan pencocokan tag pada source HTML.

2. Pengumpulan data :

Mencari studi kasus website yang akan digunakan untuk sumber data.

3. Analisis dan Desain :

Tahapan ini adalah tahapan yang meliputi analisis terhadap *Instance-based Data Extraction* kemudian akan dilakukan perancangan aplikasi untuk mengekstraksi data dari halaman web.

4. Implementasi :

Tahap ini adalah tahap pembangunan aplikasi. Pada tahap ini dibangun perangkat lunak yang akan digunakan sebagai antar muka untuk melakukan proses ekstraksi pada konten halaman web.

5. Testing :

Pada tahap ini akan dilakukan pengujian terhadap perangkat lunak yang telah dibangun, apakah sudah bekerja dengan benar atau belum. Pengujian dilakukan dengan mensimulasikan proses ekstraksi data pada suatu website.

6. Analisis hasil :

Analisis faktor-faktor yang mempengaruhi proses ekstraksi oleh *Instance-based Data Extraction* dari website.

7. Pengambilan kesimpulan dan penyusunan laporan tugas akhir :

Pengambilan kesimpulan dari hasil analisis yang telah dilakukan pada tahap sebelumnya untuk kemudian disusun laporan terhadap analisis yang telah dilakukan.

5. Kesimpulan dan Saran

5.1. Kesimpulan

Berdasarkan analisis dan pengujian terhadap sistem IDE yang telah dilakukan maka dapat diambil beberapa poin kesimpulan sebagai berikut:

1. Implementasi algoritma IDE menerapkan tiga proses utama yaitu: *page labeling, similarity measure* dan
2. Untuk menghasilkan *pattern* ekstraksi yang tepat, harus berdasarkan kepada *pattern target item* yang akan diekstrak. Pengambilan *range node* sebanyak satu template akan memberikan hasil ekstraksi yang sesuai dengan keinginan user.
3. Ketepatan nilai *range node* sangat mempengaruhi performansi sistem secara keseluruhan. Jika *node* yang diambil semakin mendekati *pattern target item* maka performansi akan semakin baik.
4. Sistem IDE memiliki performansi yang lebih baik dalam mengkestrak website berita dibandingkan dengan website daftar produk karena website berita memiliki *pattern target item* yang lebih sederhana.
5. Terdapat kasus tertentu dimana sistem IDE tidak dapat mengekstrak website jenis daftar produk walaupun range node yang diambil tidak melebihi *pattern target item* .
6. Pengukuran nilai recall didasarkan pada perbandingan data hasil ekstraksi yang relevan dari sistem dengan data relevan pada halaman web, sedangkan nilai precision didapat dari perbandingan data hasil ekstraksi yang relevan dari sistem dengan seluruh data yan berhasil diekstrak oleh sistem.
7. Sistem IDE dari hasil pengujian dengan nilai default yang ditentukan menghasilkan recall 95%, precision 79,6% untuk untuk website layanan produk, dan recall 100%, precision 90% untuk website berita.

5.2. Saran

1. Proses *page labeling* membutuhkan tampilan yang dapat memberikan info *pattern target item* secara tepat, sehingga penentuan no node dan range node lebih mudah.
2. Dibutuhkan pengembangan sistem IDE untuk mengidentifikasi paragraf dalam halaman web.
3. Pengambilan halaman web lebih baik jika dilakukan secara online.

Daftar Pustaka

[1] Bing Liu, Robert Grossman, and Yanhong Zhai, *Mining Data Records in Web Pages*, University of Illinois at Chicago, 2003, http://www.cs.uic.edu/~liub/publications/kdd2003-dataRecord.pdf,

[2] Bing Liu, *Web Content Mining*, University of Illinois at Chicago, 2005, http://www.frenchlane.com/Web-Content-Mining-4.pdf,

[3] Bing Liu, and Yanhong Zhai, *Extracting Web Data Using Instance-Based Learning*, University of Illinois at Chicago, 2007.

[4] C-H. Chang, S-L. Lui, S-L., C. Pu, *IEPAD: Information Extraction based on Pattern Discovery*, WWW-10, 2001, http://www10.org/cdrom/papers/pdf/p223.pdf,

[5] Cris Yang, *SEG5120-Web Mining*, http://www.se.cuhk.edu.hk/~seg5120/note/Lec%201%20Web%20Mining.pdf

[6] David Buttler, Ling Liu, and Calton Pu, *A Fully Automated Object Extraction System for the World Wide Web*, Georgia Institute of Technology, 2003

[7] Djoko Tri W, *Konsep Data Mining dan Implementasi (penerapan)*, ITB, 2009

[8] Kristina Lerman, Craig Knoblock, and Steven Minton, *Automatic Data Extraction from Lists and Tables in Web Sources*, Univ. of Southern California, 2003.

[9] Pramintya Purnama, Hendrawan, *Ekstraksi Informasi Pada Halaman Web dengan Memanfaatkan Mining Data Record*, Institut Teknologi Telkom, 2008.

[10] Ricardo Baeza-Yates, and Berthier Riberio-Neto, *Modern Information Retrieval*, ACM press, 1999.

[11] Dr. Spiros Sirmakesis, *Web Mining Past, Present and Future*, Computer Technology Institute, 2003, http://nemis.cti.gr/lc03/files/sirmakessis_2.pdf,

[12] http://en.wikipedia.org/wiki/Web_mining didownload pada tanggal 19 Maret 2010.