Extraction of Reliable Information from the Web

Cherry Soe, Thandar Lwin
University of Computer Studies, Yangon
cherrysoe2011@gmail.com

Abstract

Information extraction is one of the methods to retrieve information from complex web pages. With the use of multiple algorithms, intelligence, knowledge base, knowledge acquisition and filtering, people nowadays can benefited with the use of information extraction. Such application has been applied in several dimensions, such as new transcripts, insurance information, and weather reports. This proposed system extracts required laptop data from relevant web pages and convert them into a standard database. This paper uses STALKER algorithm to generate the rules for extracting the laptop information. The extracted data are matched and recognized with built in keyword and entity tables using Named Entity Recognition (NER). And then, the system produces the required extracted information. By using this system, the user can get the meaningful laptop information and it also provides the user with easy access and time saving.

Keywords: Information Extraction, Named Entity Recognition, Web mining.

.

1. Introduction

The rapid expansion of the web is causing the constant growth of information; it is becoming increasingly difficult to locate useful information. Although directories (such as Yahoo!) and search engines (such as Google) can provide some assistance, they are from perfect.

IT specialists have been trying to perfect user's needs by developing algorithms, systems, and application to provide the user with the most reliable information. Information extraction is one of the approaches to fill up a gap to overcome all the difficulty.

This paper proposes the development of information extraction application for laptop computer from IT web sites. The downloaded IT web sites are kept in folders to extract the reliable information. The keywords and entities are identified firstly. This proposed system uses STALKER algorithm to generate the rules. To recognize the extracted data, NER is used.

This proposed system intends for users who need to search for the information for the configuration and price of a laptop computer, before shopping on the website or go personally to the computer retail shop.

The main objective of this system is:

- To study the efficiency of information extraction from the web
- To understand the information extraction methods
- To study the conversion to standard database from the extracted data of the web pages
- To provide for the purpose of ease accessing and time saving

The paper is organized as follows. Section 2 provides the related work for the system. Section 3 describes the background theory of the system. Section 4 presents the proposed system. Section 5 shows the implementation of the system. Section 6 displays the experimental results. Section 7 describes the limitation of the system and section 8 is the conclusion of the system.

2. Related Work

Cohen and Fun [2] present a method for learning page-independent heuristics for IE from Web pages. However they require as input a set of existing wrappers along with the pages they correctly wrap.

Cohen et al. [3], also present one component of a larger system that extracts information from multiple sites. A common characteristic of both the aforementioned approaches is that they need to encounter separately each different markup structure during training. In contrast to this approach, we examine the viability of trainable systems that can generalize over unseen sites, without encountering each page's specific structure.

An IE system that exploits shallow linguistic pre-processing information is presented in [6]. However, they generalize extraction rules relying on lexical units (tokens), each one associated with shallow linguistic information, e.g., lemma, part-of-speech tag, etc. We generalize rules relying on named entities, which involve contiguous lexical units, and thus providing higher flexibility to the wrapper induction algorithm.

An ontology-driven IE system from pages across different sites is presented in [3]. However, they rely on hand-crafted (provided by an ontology) regular expressions, along with a set of heuristics, in order to identify single-slot facts within a document. On the other hand, we try to induce such expressions using wrapper induction.

This paper uses STALKER algorithm to generate the rules. Named Entity Recognition (NER) is also used to recognize the extracted data.

3. Background Theory

3.1 Information Extraction

Information extraction uses techniques different from information retrieval (IR) to obtain a higher degree of knowledge from textual information sources. The information obtained is structured and relatively easy to analyze to provide directly useful information. Information Extraction (IE) is the identification and extraction of instances of a particular class of events or relationships in a natural language text and the transformation into a structured representation (e.g. a database) [8].

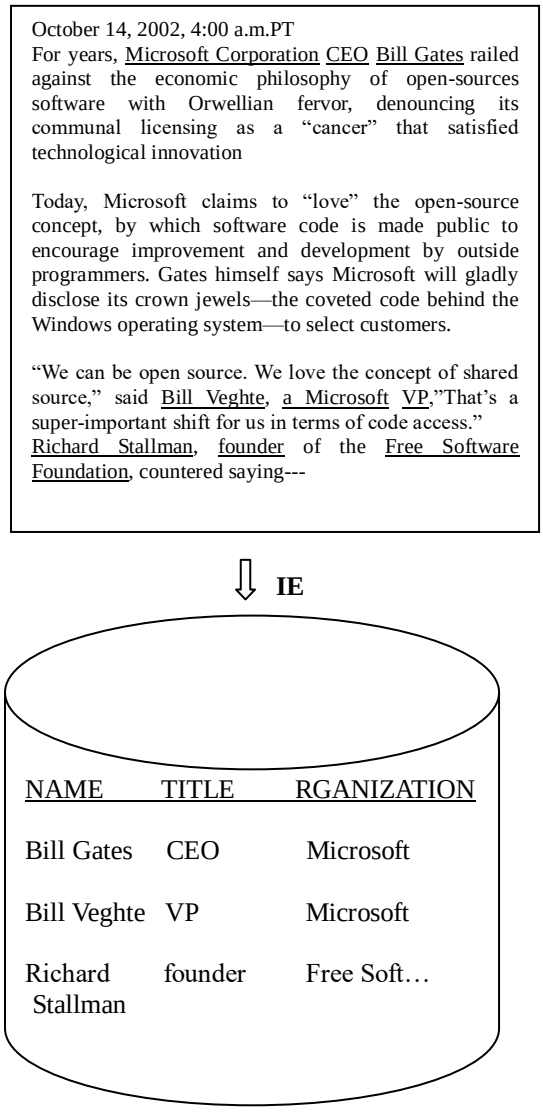

Figure.1. Information Extraction

Figure.1 displays extraction of information from a web page and transforms a standard database.

3.1.1 Wrapper

Wrapper is a procedure to extract all kinds of data from a specific source. First find a vector of strings to delimit the extracted text.

If the web page format is as follows,

<**HTML**><**TITLE**> Country Codes </**TITLE**>
<**BODY**><**B**> Congo </**B**><**I**> 242 </**I**><**BR**>
<**B**> Spain </**B**> <**I**> 34 </**I**> <**BR**>
<HR> <B> END </B> </BODY> </HTML>

The objective of a wrapper is to extract pair (country, codes), we find a vector of strings (<B>, </B>, <I>, </I>) to distinguish left and right of the extracted text.

3.1.2 Wrapper Induction Systems

Independently of the traditional IE community, the wrapper generation field appeared from the necessity of extraction data from multiple web based sources. Wrapper Induction (WI) system constructs extraction rules called wrappers to extract information from web pages. The wrapper induction systems generate delimiter-based rules that do not use linguistic constraints. In order to facilitate the comparison between the several of extraction patterns, all three types of rules discussed below are described.

WIEN is the first wrapper induction system, and it generates extraction rules, that it uses only delimiters that immediately precede and follow the actual data. WIEN assumes that there is a unique multi-slot rule that can be used for all documents, and does not allow the use of semantic classes.

Soft Mealy is a wrapper induction algorithm that generates extraction rules expressed as finite state transducers. It allows both the use of semantic classes and disjunctions, which are especially useful when the documents contain several formatting convention or various orderings of the items of interest.

STALKER is a wrapper induction system that performs hierarchical information extraction. STALKER introduces the Embedded Catalog Tree (ECT) formalism to describe the hierarchical organization of the documents. STALKER generated one extraction rule for each node in the tree, together with an additional iteration rule for each List node. The extraction process is performed in a hierarchical manner.

With the growth of the amount of online information, the availability of robust, flexible IE systems will become a stringent necessity. Depending on the characteristics of application domains, today's IE systems use extraction patterns based on one of the following approaches: syntactic / semantic constraints, delimiter-based rules, or a combination of both.

3.1.3 Named Entity Recognition

Named Entity Recognition (NER) is an important step in the information extraction task. The overall information extraction task to extract facts in the form of multi-place relations and NER provides the entities that fill the relevant fields.

3.2 STALKER Wrapper Induction System

STALKER is capable of extracting information from pages with tabular organization of their content, as well as pages with hierarchically organized content. The hierarchically content is constructed manually in a catalogue tree structure. Using the catalogue tree as a guide, the extraction in a given page is performed [7]. Using ECT has two major advantages. First, STALKER was the only IE inductive system that can extract data from documents that contain arbitrarily complex combination of embedded lists and items. Second, as each item is extracted independently of its siblings in the ECT, the various orderings of the items does not require one rule for each existing permutation of the items to be extracted.

The STALKER algorithm is used to generate the extraction rules. The STALKER Algorithm is shown in Figure.2.

```
Use a sequential covering algorithm

STALKER (examples)
    Set setRule be empty
    While there are more examples
        Get a disjunct D by learning examples
        Remove all examples covered D
        Add D into setRule

Return setRule
```

Figure.2. STALKER Algorithm

4. Proposed System

In this paper, the laptop information extraction is done as the proposed system. The overview of the proposed system is shown in Figure.3. Firstly, the system identifies laptop entities such as laptop name, processor, memory, hard disk, display, wireless, drive, price as shown in Table 1.

Table1. Entity Table

EID	Entity Name
1	LName
2	Processor
3	Memory
4	HDC
5	Display
6	Wireless
7	Drive
8	Price

The keywords are identified to recognize which name entity the extracted data using extraction rules see in Table 2.

Table2. Keyword Table

KeywordID	Keyword	EID
1	Acer	1
2	Asus	1
3	Dell	1
4	Samsung	1
5	Toshiba	1
6	Sony	1
7	IBM	1
8	Compaq	1
9	Pentium	2
10	AMD	2
11	Atom	2
12	Core	2
13	RAM	3
14	Memory	3
15	DDR	3
16	HDD	4
17	HD	4
18	GB	4
19	“	5
20	Display	5
21	Inch	5
22	DVD RW	7
23	CD	7
24	DVD	7
25	Wireless	6
26	£	8
27	$	8

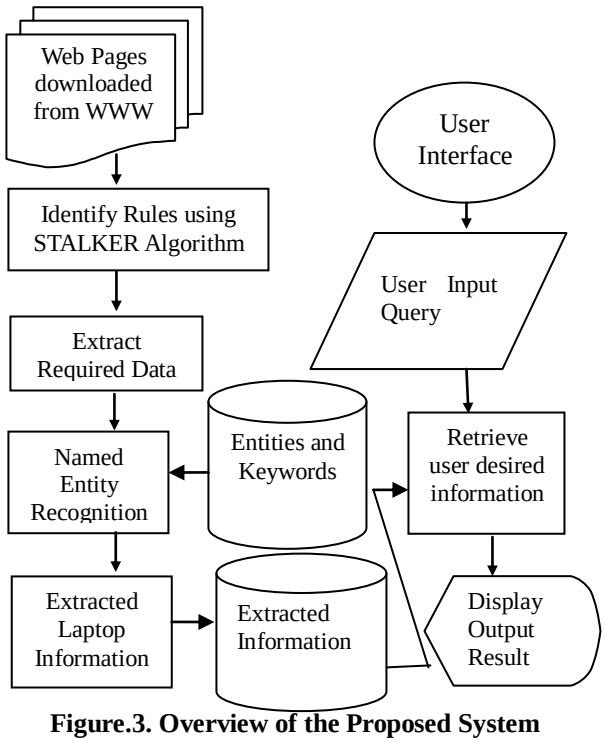

Figure.3. Overview of the Proposed System

The system accepts the folder that has laptop web pages and other pages that do not have the laptop information downloaded from the World Wide Web. From these pages, the system will extract only the relevant information from the qualified (laptop) web pages.

```
<img alt="" src="page4%28site%20%29_files/
free-ability-office-software.jpg"
width="378"height="101"> <br>
<table width = “180” cellpadding = “0”
        cellspacing = “0” border = “0”>
    <tr>
        <td class = “productlisting -data” style
            = “text-align: left”; align = “left”>
          <b> Acer i5535S </b></td>
    </tr>
    <tr>
        <td class = “boxText” style= “text-
        align: left”; align = “left”>
          <span class= “boxText”> AMD
          Athlon QL60 1.9 GHz
                </span>
          < span class = “boxText”> 160 GB
                </span>
          < span class = “boxText”> Massive 3
                Gb RAM </span>
          < span class = “boxText”> DVD/RW
                </span>
          < span class = “boxText”> 15.6”
                CrystalBrite TFT </span>
          < span class = “boxText”> Wireless
                </span>
        </td>
    </tr>
    <tr>
        <td class = “boxText” valign = “top”>
          <b> our price: </b></td>
        <td class = “boxTextRed”>
          <b> £ 249.00 </b></td>
  </tr>
 </table>
```

Figure.4. Sample HTML source code of a laptop web Page

STALKER introduces the Embedded Catalog Tree (ECT) formalism to describe the hierarchical organization of the documents. The EC description of a page is a tree-like structure in which the leaves represent the information to be extracted. Figure.5**.** displays the EC description of the laptop web pages. At the top level, each page is a list of laptop descriptions. Each laptop description is many tuples.

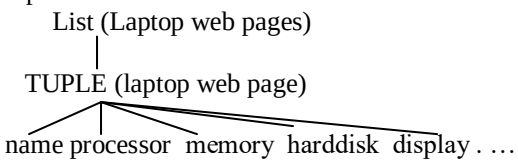

Figure.5. EC description of laptop web page

The system loads HTML tags as tree-like structure and examine each node. If the node has child nodes, the system will save the pattern as RULE and follow hierarchy. The system will skip to inner node when it found similar RULE again. When the system found no child node, the data is extracted.

The laptop description presented in Figure.4**.** In order to identify the beginning of the laptop name within a laptop description, we can use the rule

R1 = SkipTo (<table><tr><td><b>)

The rule R1 has the following meanings: start from the beginning of laptop description and skip everything until finding <table><tr><td><b> tags, and extract behind these tags the data.

To identify the beginning of the processor within a laptop description, we can use the rule

R2 = SkipTo (<table><tr><td>)

The rule R2 has the following meanings: start from the beginning of laptop description and skip everything until finding <table><tr><td> tags and extract behind these tags the data.

To recognize the extracted data, named entity recognition is used. The each extracted data is matched with the each predefined keyword and then the extracted data is entered into the relevant fields.

The system displays the extracted laptop information as table form and stores the extracted laptop information into database for querying laptop information by user.

5. Implementation of the system

The system is implemented for giving the reliable laptop information from web. The system is applied for laptop category information records. This system is necessary for buyers who have extensive choice of feature, brands, and prices. The system can support the buyer’s requirement with variety of laptop lists so that he/she can make choice that meets his/her design and other supported information.

The main frame of the system which contain two process buttons. The first one is “Extract information from web pages” button and “Search from extracted information” button.

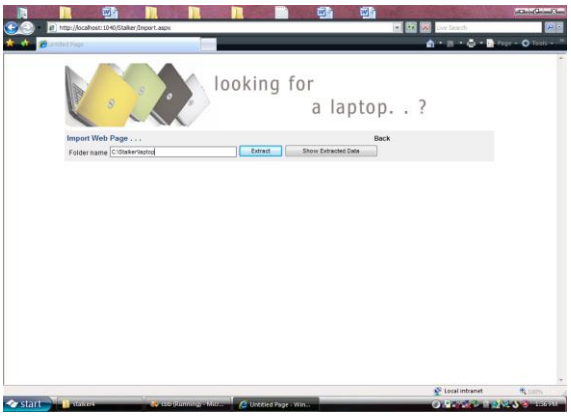

Figure.6. Input the folder that have web pages to extract laptop information

When the user selects “Extract information from web pages” button, the next interface will be appeared. The user input the folder that has the laptop web pages and other pages that do not contain the laptop information. And then the user clicks the extract button as shown in Figure.6.

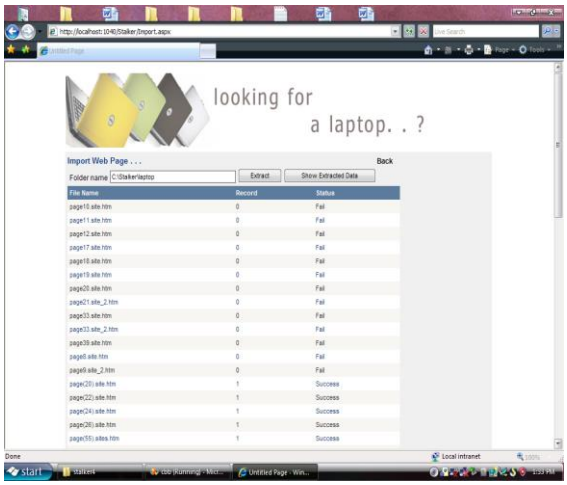

Figure.7. Showing the fail status and the success status for the extracted pages.

The system shows that the pages with fail status are irrelevant pages. Success status show the web page has one or more laptop data, seen in Figure.7.

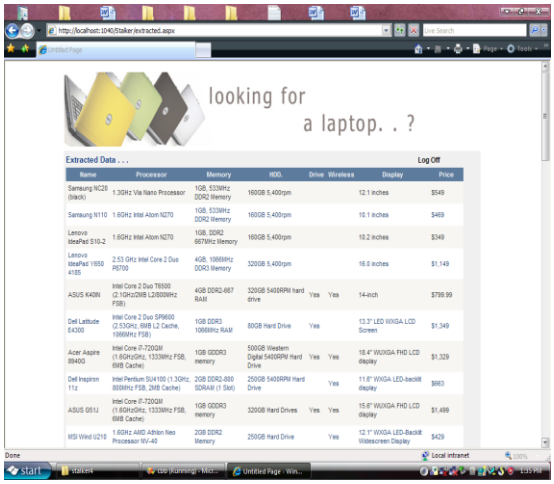

Figure. 8. The extracted results from the laptop web pages

When the user chooses “Show Extracted Data” button, the system displays the extracted laptop information. If the user click save button, the system will save the extracted laptop information into the database shown in the Figure.8.

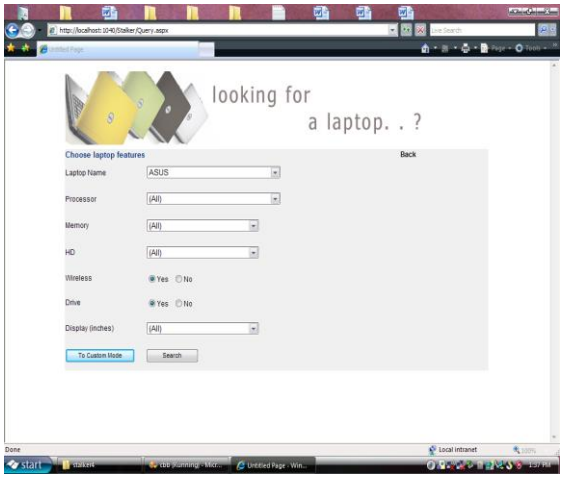

Figure. 9. Input user interface for data search

The system also provides search features for querying laptop information. The user selects laptop name as Asus, wireless as Yes, and drive as Yes as shown in Figure. 9.

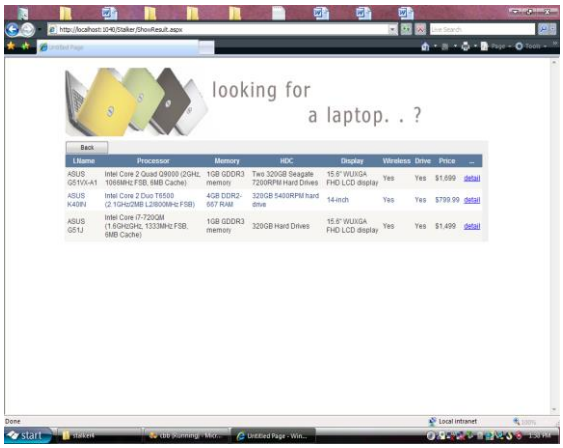

Figure. 10. Output data from data search

The system displays user desired laptop information by retrieving from database seen in Figure. 10.

6. Experimental Results

Experiments are carried out in the domain of laptop web pages. This system can give laptop information for user, by extracting with rules applying STALKER algorithm. Table3. shows the experimental results for each web site.

$$\text{Precision} = \frac{\text{Correctly Extracted Information}}{\text{Total Extracted Information}}$$

$$\text{Recall} = \frac{\text{Correctly Extracted Information}}{\text{Total Number of Correct Information}}$$

Table 3.Experital Results for each web site

Websites	Number of pages	Extracted Information		
		Precision	Recall	Errors
www.cnet.com	80	75	92.6	7.4
www.notebookcomputer.com	100	80	94.1	5.9
www.studentcomputer.com	90	77.7	93.3	6.7

7. Limitation

Some limitations exist in this system. This system can only extract for Laptop Web Pages which are allowed to extract only the Laptop name, processor, memory, hard disk, drive, display, wireless, and price. This system can extract from table format web pages. Moreover, these web pages must be written in English. If input documents have errors or are irrelevant it cannot extract right information.

8. Conclusion

This proposed system uses the STALKER wrapper induction method to extract the reliable information from laptop web pages. The STALKER and NER methods support the extraction process and get the required data. The system can be able to extract the required laptop information from the web pages to build the database for the purpose of ease in accessing and time saving. This system is developed in a way to provide the user with easy-to-use. The system also has efficient scanning of large volume of web documents and the discovery of significant facts.

9. References

[1] Ciravegna, F., "Adaptive Information Extraction from Text by Rule Induction and Generalization." In Proceedings of the 17th IJCAL Coference. Seattle (2001).

[2] Cohen, W., Fan, W., "Learning page-independent heuristics for extracting data from Web pages." In the Proceedings of the 8th international WWW Conference (WWW-99).Toronto, Canada (1999).

[3] Cohen, W., Hurst, M., Jensen, L., "A Flexible Learning System for Wrapping Tables and Lists in HTML Documents." Proceedings of the 11 International WWW Conferences. Hawaii, USA (2002).

[4] Davulcu, H., Mukherjee, S., Ramakrishman, I.V., "Extraction Techniques for Mining Services from Web Sources," IEEE International Conference on Data Mining, Maebashi City, Japan (2002).

[5]Georgios Sigletos ,Georgios Paliouras,Constantine D. "Mining Web sites using wrapper induction, named entities and post processing," Spyropoulos, Michalis Hatzopoulos, Institute of Informatics and Telecommunications, NCSR.(2003).

[6] Hongfei Qu, "Wrapper Induction: Construct wrappers automatically to extract information from web sources" Computing Science Department, Simon Fraser University, CMPT 882 Presentation, March 28, 2001.

[7] Jiying Wang, "Information Discovery, Extraction and Integration for the Hidden Web," Department of Computer Science University of Science and Technology, Clear Water Bay, Kowloon Hong Kong (2003).

[8]MUC-7, http://www.itl.nist.gov/iaui/894.02/related_projects/muc.

[9] Muslea, I., Minton, S., Knoblock, C.,"Hierarchical Wrapper Induction for Semistructured Information Sources" from Journal of Autonomous Agents and Multi-Agent Systems, 4:93-114 (2001)