A Proof-of-Concept of D^3 Record Mining using Domain-Dependent Data

Yeong Su Lee[1], Michaela Geierhos[1], Sa-Kwang Song[2], and Hanmin Jung[2]

[1] Center for Information and Language Processing, University of Munich, Germany
{yeong|micha}@cis.uni-muenchen.de
[2] Korea Institute of Science and Technology Information (KISTI), Daejeon, Korea
{esmallj|jhm}@kisti.re.kr

Abstract. Our purpose is to perform data record extraction from online event calendars exploiting sublanguage and domain characteristics. We therefore use so-called domain-dependent data (D^3) completely based on language-specific key expressions and HTML patterns to recognize every single event given on the investigated web page. One of the most remarkable advantages of our method is that it does not require any additional classification steps based on machine learning algorithms or keyword extraction methods; it is a so-called one-step mining technique. Moreover, another important criteria is that our system is robust to DOM and layout modifications made by web designers. Thus, preliminary experimental results are provided to demonstrate proof-of-concept of such an approach tested on websites in the German opera domain. Furthermore, we could show that our proposed technique outperforms other data record mining applications run on event sites.

1 Introduction

There are numerous web sites providing large databases containing information such as yellow page listings or event calendars. Current approaches to data record mining [1, 2] exploit the structured character of HTML documents. For this purpose, two or more similar web pages have to be compared in order to extract the corresponding data records. These systems often expect preclassified web pages as input [2]. Based on the fact that data records are dynamically generated from a back-end database, some applications like MDR [3] try to reconstruct the given web page benefiting from the regularities of HTML structure. Hereby, the main focus is to determine iterations of HTML tag sequences by using the DOM tree representation of a web page. Zheng et al. [4] try to extract records represented by the so-called "broom" structures. "In this approach a set of training pages are converted to DOM trees by an HTML parser. Then, semantic labels of a specific extraction schema are manually assigned to certain DOM nodes to indicate their semantic functions. Based on these labels, a broom-extraction (...) algorithm can be applied on each DOM-tree" [5, p. 163]. Although they achieve high values in precision and recall, the training pages have to be manually annotated because most wrapper-induction systems expect such user-generated input formats.

Due to the loose strictness of HTML, others [6–8] try to exploit the visual information provided by a web browser by using rendering techniques for data record mining. They apply these methods on the displayed query results in order to determine the record boundaries. Even flat and nested data records can be extracted by Visual Structure based Analysis of web Pages [8] based on some heuristics [7]. Although rendering methods achieve good results, they have one big drawback: They require a web browser, which correctly displays the investigated web page, to determine some typical visual cues of a data region (e.g. size, background color, icons, font colors).

Regardless of the technique used, all methods have one point in common: Current approaches to data record mining disregard any language-specific information dependent on the application domain. We observed that existing data record mining techniques are not satisfactory enough for specialized search purposes limited to restricted domains across websites. The success of event calendar search highly depends on language-specific trigger words indicating at least some date. We therefore propose a novel method for data record mining on demand. Browsing restricted domains allows us to define some key words, e.g. weekdays, nested in the data records of event sites. By means of limited vocabulary, we are able to analyze the document's HTML structure and locate the corresponding data record boundaries. Our technique is quite robust in variability of the DOM, upgradeable and keeps data up-to-date.

The paper is structured as follows. In Section 2, we present our data mining technique. Section 3 evaluates the proposed method. In Section 4, we summarize our work and finally highlight future research directions.

2 The proposed technique

After retrieving a large website, each web page has to be classified into pages with event calendars or without, depending on its key expressions[3]. If the page contains at least two or more key expressions, then the search for event calendar records (ECR's)[4] will start. Otherwise, the page will be skipped. Thus, the classification of pages containing ECR's can be performed without the help of machine learning algorithms or keyword extraction methods.

We now present the two steps of our approach:

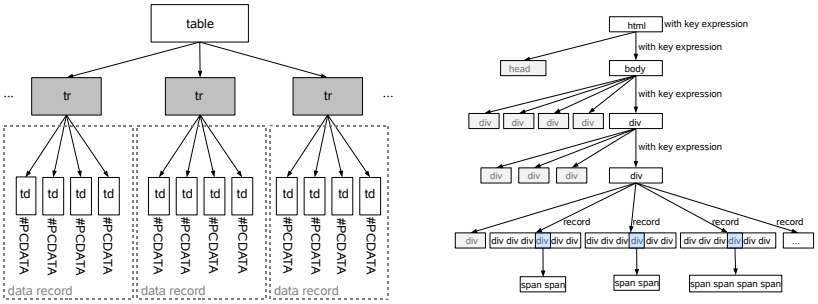
[3] An instance of a feature set that classifies a data record and can be described by regular expressions or string variants is called key expression. Please note that the selection of a key expression highly depends on the record type and on the domain respectively. For example, within a shopping record, the key expression may be some price information, and within a computer description, it may be the CPU type. We therefore speak of domain-dependent data described by its corresponding key expressions based on a limited vocabulary.

[4] An event calendar record (ECR) is primarily a data record which provides information on event details like event title, event location, event date, etc.

1. First, we create the DOM tree of a selected web page in order to exploit its HTML structure (cf. Section 2.1)
2. Secondly, we assume that there is only one smallest maximum data region for the ECR's [3, 8–10] and it corresponds to only one HTML tag region. The smallest maximum data region can be determined by a top-down traversal of the tree using key expressions. It is predictable that there must be two or more key expressions in one HTML tag region of the tree. Otherwise, this tag region will be cut off from the DOM tree.

2.1 Exploiting the structure of ECR's within the DOM tree

Each website has its own distinct method of presenting information. Therefore, the high variability observed in HTML structure should be taken into account. However, the number of possible tag combinations which can be considered for event calendar records (ECR's) is very limited.



(a) One record under one tag region (b) Sample core tree structure

Fig. 1. Different types of event calendar records

The following types of ECR's according to their tree structure have been registered and were classified as follows:

(a) One single record under one node (cf. $\langle tr \rangle$ in Fig. 1(a))
(b) Each record consists of a set of children nodes with the same parent node.

In Figure 1(a), we act on the assumption that all data records are siblings among each other with their own parent nodes, all containing the same value (e.g. $\langle tr \rangle$). If this tag region contains some key expression (e.g. weekday) and other event-related information, it will be selected as ECR.

In Figure 1(b)), each ECR consists of a set of children nodes belonging to the same parent node. We can thereby distinguish between three structure types of data records (HTML tag regions) depending on the co-occurrence of tag attributes and values.

1. repetition of an HTML tag, e.g. ⟨*div*⟩, with non-recurring attributes within the data records, including their text values,
2. repetition of an HTML tag, e.g. ⟨*div*⟩, with non-recurring attributes within the data records and incomplete attribute-value-pairs (Fig. 2),
3. missing both tag attribute and value.

Among these, case (1) is really rare and (3) can sometimes happen, but in practice, case (2) occurs quite frequently.

In Figure 1(b), we showed that the key expression is inherited to only one HTML tag node (⟨*html*⟩ → ⟨*body*⟩ → the 5^{th} ⟨*div*⟩ → the 4^{th} ⟨*div*⟩), and all records are the children of this one single node (cf. ⟨*div class="content"*⟩ in Figure 2). When we zoom in and look at the record structure in detail, each record is composed of six ⟨*div*⟩ tags and its corresponding attributes: *"kalendariumtag"*, *"kalendariumdatum"*, *"kalendariumuhrzeit"*, *"kalendariummitte"*, *"kalendariumpreise"* and *"kalendariumlinie"*.

As shown in Figure 2, the text values (ε) are missing for the first two mentioned HTML tag attributes in the one record, but are filled with #PCDATA "A1" and #PDCATA "B1" in the preceding record. We therefore resolve such coreferences by linking the text values of the same attributes in successive records.

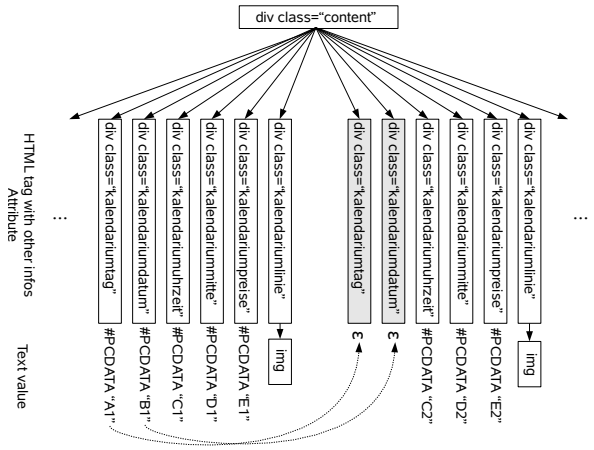


Fig. 2. Tag iteration of ⟨*div*⟩ with attributes and missing text values within an ECR

But one problem that still remains to be solved is how to decide where the record starts and where it ends: The boundary between records can be determined by comparing the bordering tag attributes. Based on the assumption that the key expression, e.g. weekday, is placed in first position (cf. #PCDATA "A1" in Figure 2), then we have two possibilities: We can go forward or backward to recognize the record boundaries. If we move forward, the same tag attribute

will recur after six steps. That way, we learn that one record consists of six tag attributes. However, we do not know yet where the record begins. In order to solve this problem, we go back until we find a tag attribute totally different from the six common attributes in Figure 2. Now we can initialize the starting points for all records embracing six tag attributes each.

2.2 Computing the tag similarity

Assuming that some tag attributes are varying for the same type of event information because of layout issues, then we compute their tag similarity *sim* by using Dice's coefficient: $sim = \frac{2|X \cap Y|}{|X|+|Y|}$
In the particular case here considered, we have two tag attributes for *date* and *weekday* which differ in background color, but both specify the day the event takes place. Thus, we consider them as one single information bit.

⟨*td class="verd12" width="56" align="center" valign="top" bgcolor="#ffffff"*⟩
⟨*td class="verd12" width="56" align="center" valign="top" bgcolor="#cc0000"*⟩

Despite their attribute differences we can assign both tags to the same type of information by calculating their tag similarity. We therefore separate the corresponding attribute-value pairs from the HTML tag: For this example, we extract {*"class", "width", "align", "valign", "bgcolor"*} as attribute set and {*"verd12", "56", "center", "top", "#ffffff"*} as value set of the first tag. Moreover, we can observe that for the second tag, there is only one difference concerning the value of *"bgcolor"* which is *"#cc0000"*. Then, we apply the function *sim* where X is the set of attributes and values of the frist tag and Y of the second tag: In our example, |X| is 10, as is |Y| because the attribute set and value set contain five elements each. Consequently, $|\{X\} \cap \{Y\}|$ equals 9.

$$sim = \frac{2 \cdot 9}{10 + 10} = \frac{18}{20} = 0.9$$

In order to compare two tags, their position within one event record has to be the same, otherwise we cannot compute their tag similarity. Furhtermore, the similarity threshold can be adjusted by the help of some heuristics.

3 Experimental evaluation

oper-frankfurt.de To evaluate the quality of the proposed record mining technique from arbitrary websites, we concentrate our case study on websites of German opera and theater houses. Our test set consists of 20 event calendar pages randomly retrieved from websites of opera houses (e.g. `oper-frankfurt.de`, `bayerische.staatsoper.de`, `staatsoper-berlin.org`, `semperoper.de`). We achieved a recall of 93.81% on the test data.

4 Upgradable features and future work

One presumption is that our key expression driven record mining technique expects a valid HTML page for the DOM tree construction. If the tree cannot be built up, we will use some open source tools for correction purposes. We must admit that our approach is not able to reconstruct the DOM tree of web pages with no closing HTML tags. Thus, our method disregards totally nested record structures in order to protect itself of analyzing ECR's nested in other ECR's ad infinitum.

So far we use hand-coded key expressions for the opera domain, but we work on learning such rules automatically. By measuring the similarity of content strings or tag regions, we will figure out the best candidates for domain-specific key expressions. Of course, they are language-dependent and we have to expand them to other languages apart from German.

Moreover, until now we concentrated on a very special domain – event calendars of opera houses. It could be interesting to adopt this technique to other domains dealing with different event information (e.g. sports, exhibitions, fairs).

References

1. Arasu, A., Garcia-Molina, H.: Extracting Structured Data from Web Pages. In: Proceedings of the 2003 ACM SIGMOD international conference on Management of data, San Diego, California, USA (2003) 337–348
2. Crescenzi, V., Mecca, G., Merialdo, P.: RoadRunner: Towards Automatic Data Extraction from Large Web Sites. In: Proceedings of the 27th VLDM Conference, Rome, Italy (2001)
3. Liu, B., Grossman, R., Zhai, Y.: Mining data records in web pages. In: KDD '03: Proceedings of the ninth ACM SIGKDD international conference on knowledge discovery and data mining, Washington D.C., USA (2003) 601–606
4. Zheng, S., Song, R., Wen, J.R., C. Lee Giles: Efficient Record-Level Wrapper Induction. In: CIKM'09, Hong Kong, China (2009) 47–55
5. Ashok Kumar, R., Rama Devi, Y.: Efficient approaches for record level web information extraction systems. International Journal of Advanced Engineering & Application **2**(1) (January 2011) 161–164
6. Cai, D., Yu, S., Wen, J.R., Ma, W.Y.: Extracting Content Structure for Web Pages based on Visual Representation. In: Web Technologies and Applications: 5th Asia-Pacific Web Conference, APWeb 2003, Xian, China (2003)
7. Algur, S.P., Hiremath, P.S.: Visual Clue Based Extraction of Web Data from Flat and Nested Data Records. In: International Conference on Management of Data (COMAD 2006), Dehli, India (2006)
8. Hiremath, P.S., Benchalli, S.S., Algur, S.P., Udapudi, R.V.: Mining Data Regions from Web Pages. In: International Conference on Management of Data (COMAD 2005), Hyderabad, India (2005)
9. Liu, W., Meng, X., Meng, W.: Vision–based Web Data Records Extraction. In: Ninth International Workshop on the Web and Databases (WebDB 2006), Chicago, USA (2006) 20–25
10. Zhai, Y., Liu, B.: Web Data Extraction Based on Partial Tree Alignment. In: WWW2005, Chiba, Japan (2005)