SILA: a Spatial Instance Learning Approach for Deep Web Pages

Ermelinda Oro
Institute of High Performance Computing and Networking
of the Italian National Research Council
Via. P. Bucci, 41/C, University of Calabria
87036, Rende (CS), Italy oro@icar.cnr.it

Massimo Ruffolo
Institute of High Performance Computing and Networking
of the Italian National Research Council
Via. P. Bucci, 41/C, University of Calabria
87036, Rende (CS), Italy ruffolo@icar.cnr.it

ABSTRACT

Deep Web pages convey very relevant information for different application domains like e-government, e-commerce, social networking. For this reason there is a constant high interest in efficiently, effectively and automatically extracting data from Deep Web data sources. In this paper we present SILA, a novel Spatial Instance Learning Approach, that allows for extracting data records from Deep Web pages by exploiting both the spatial arrangement and the presentation features of data items/fields produced by layout engines of Web browsers in visualizing Deep Web pages on the screen. SILA is independent from the internal HTML encodings of Web pages, and allows for recognizing data records in pages having multiple data regions in which data items are arranged by many different presentation layouts. Experimental results show that SILA has very high precision and recall and that it works much better than MDR and ViNTs approaches.

Categories and Subject Descriptors
[Knowledge Management (KM)]

General Terms
Algorithms, Experimentation

Keywords
Web Information Extraction, Deep Web, Instance Learning, Web Wrapping

1. INTRODUCTION

The Deep Web is the part of the Internet that is not accessible by conventional search engines. Deep Web pages are dynamically generated from databases in response to queries submitted via search forms filled in by keywords. The Deep Web continue to grow as organizations and companies, operating in fields like e-government, e-commerce and social networking, make available their large amounts of data by providing Web-access facilities to their databases. Consequently, there is a constant high interest in efficiently extracting data from Deep Web data sources.

A large body of work on approaches for extracting data from Deep Web sources is already available in literature. Already existing approaches can be classified, for the scopes of this paper, in two main groups: (i) approaches that mainly use the internal representation of Deep Web pages [1, 5, 6], and (ii) approaches that exploit the visual appearance of Deep Web pages [2, 7]. Approaches in both groups are still limited in many aspects. Methods based on the internal structure of Deep Web pages suffer of the complexity of today Web pages encodings. In fact, they need to be updated for facing the adoption of new standards and tags. In particular, the growing adoption of scripts and CSS style sheets, for presenting data to human users, makes Web pages more complex than ever. Even if the layout of information on Web pages provides visual cues that help human readers to make sense of Web pages contents, current approaches that exploit the visual appearance of Web pages adopt page segmentation algorithms and heuristics that require specific and, sometime, complex tunings.

In this paper we present a novel spatial instance learning approach for Deep Web pages that exploits both the spatial arrangement and the presentation features of data records and data items/fields produced by layout engines of Web browsers. Main contributions are: (i) The definition of a novel spatial data model for Web pages named PDOM, i.e. Positional Document Object Model. (ii) A novel visual similarity measure that computes similarity between sets of PDOM nodes by exploiting their spatial and presentation features. (iii) The definition of an efficient and effective instance learning algorithm, based on a hierarchical clustering technique and heuristic aggregation methods, that allows for recognizing data records and data items in Deep Web pages independently from their visual arrangement.

The SILA algorithm has been tested on a dataset of 210...
Deep Web pages randomly selected from most known Deep Web sites. Experimental results have shown the effectiveness of SILA. In particular, carried out experiments have shown that SILA: (i) allows for recognizing data records and items having any spatial arrangement and spread on multiple (data) regions of a single page, (ii) has very high precision and recall and works much better than already existing and well known MDR [1], and VINTs [7] approaches.

2. A SPATIAL DATA MODEL FOR WEB PAGES

In this section we firstly introduce the notion of Positional Document Object Model (PDOM) of Web pages. Then we describe how PDOMs are created by considering both the traditional DOM representation and the spatial arrangement of Web pages obtained from layout engines of Web browsers.

A Web page can be seen as a 2-dimensional Cartesian plane on which are placed 2-dimensional objects (e.g. data records and items) surrounded by Minimum Bounding Rectangles (MBRs), as shown in Figure 1. In spatial reasoning [3], MBRs are the most common approximations of 2-dimensional objects because they need only two points for their representation in the Cartesian space. The concept of MBR is defined as follows.

**Definition 1. Minimum Bounding Rectangle.** Let $o$ be a 2-dimensional object, the minimum bounding rectangle (MBR) of $o$ is the minimum rectangle $r$ that surrounds $o$ and has sides parallel to the axes ($x$ and $y$) of the Cartesian plane. We call $x_r$ and $y_r$ the segments that are obtained as the projection of $r$ on the $x$-axis and the $y$-axis respectively. Then, each side of the rectangle is represented by the segments $(x_r,y_r)$ and $(x_r,y_r)$, where $x_r$ (resp. $y_r$) denote the infimum on the $x$-axis ($y$-axis) and $x_r$ (resp. $y_r$) denote the supremum on the $x$-axis ($y$-axis) of the segments $x_r$ and $y_r$ respectively.

**Figure 1: MBRs of DOM nodes**

Considering MBRs, directional and containment relations among 2-dimensional objects can be simply modeled. For representing directional relations we adopt the Rectangular Cardinal Relation (RCR) spatial reasoning model [3].

We define the Positional Data Object Model (PDOM) of Web pages which the proposed spatial instance learning approach is based on. A PDOM is a tree structure where:

(i) the parent-child relation represents spatial containment between nodes (PNodes) and (ii) each PNode, named positional node (PNode), represents one or more DOM nodes laid out on the screen by the layout engine of a Web browser. PNodes are equipped by both spatial (i.e. the position in which DOM nodes are visualized on the screen by the layout engine of a Web browser) and presentation features (i.e. font color, font or image size, font style, background color, borders, etc.). In order to explain why we adopt a spatial representation of Deep Web pages we point out that Web designers frequently use very involved and undergoing HTML structures for obtaining visual appearance of data records in Deep Web pages that make sense for human readers, as already discussed in [4]. Such structures pose many problems to already existing instance learning and wrapper induction techniques. At the contrary, the PDOM, allows for representing Web pages in a quite simple way. So, pages that show the same visual pattern to human readers, may have very different internal HTML encodings, as shown in Figure 2a-b. By the PDOM representation such pages can be made more easy to process for instance learning and wrapper induction approaches. Furthermore, frequently it is possible to obtain the same PDOM represetation for different pages as shown in Figure 2c-d.

**Figure 2: DOMs fragments of Web pages representing friend lists in social networks: (a) Care2.com, and (b) Bebo.com. PDOMs - (c) and (d) - representing the visual arrangement for page fragments in (a) and (b) respectively.**

It is worthwhile noting that in Deep Web pages data records are usually arranged either as lists or matrices where data items can be indifferently organized in vertical or horizontal way as depicted in Figure 2.

A human-oriented visual pattern for data records can be obtained from very different HTML encodings. The PDOM has the property to generalize different possible HTML encodings by combinations of only four different structural patterns that we call: standard, flat, nested, and non-contiguous. In the standard PDOM record pattern, each data record is represented by a single PNode which subtree constitute all data items of the data record as depicted in Figure 4a. In the nested PDOM record pattern, groups of data records have a common parent PNode, as sketched in Figure 4b. In the flat PDOM record pattern, data items of data records of a data region have the same parent PNode, as shown in Figure 4c. Finally, in the non-contiguous PDOM record pattern, groups of data items of different data records can have different parents as shown in Figure 4d.

PNodes and the PDOM are formally defined as follows.

**Definition 2. PNode.** A PNode is a 3-tuple of the form:

$$PNode = (hvalue, mbr, Style)$$

where:
• **value** is the value of the PNode. The value is either a string or an URL of an image.

• \( \text{mbr} = h( r_x, r_y, r_x, r_y) \) is the MBR (as defined in Definition 1) of the PNode. The MBR define the absolute position of the PNode on the screen.

• **Style** is the set of presentation features of the PNode.

For computing the PDOM, the SILA system embeds the Mozilla browser by exploiting the Mozilla XULRunner application framework that allows for implementing the function \( \text{mbr} \) (see Definition 1).

In the DOM there are chains of nodes that are rendered in the same MBR. For this reason, a PNode \( p \) can represents more DOM nodes. Let \( D \) be a DOM, a PDOM \( P \) is built on the base of the containment relations among the MBRs of nodes in \( D \), starting from the root node of \( D \). More in detail, for each pair of nodes \( u \) and \( v \) in \( D \), we have:

- If \( \text{mbr}(u) = \text{mbr}(v) \), then \( u \) and \( v \) are represented by the same PNode \( p \).
- If \( \text{mbr}(u) \) contains \( \text{mbr}(v) \), then \( u \) corresponds to a PNode \( p_1 \) and \( v \) correspond to a PNode \( p_2 \) and \( \text{mbr}(p_1) \) contains \( \text{mbr}(p_2) \).
- If \( \text{mbr}(v) \) contains \( \text{mbr}(u) \), then \( v \) corresponds to a PNode \( p_1 \) and \( u \) correspond to a PNode \( p_2 \) and \( \text{mbr}(p_1) \) contains \( \text{mbr}(p_2) \).

else, there exists two PNodes \( p_1 \) and \( p_2 \) (such that \( \text{mbr}(p_1) \) do not intersect \( \text{mbr}(p_2) \)).

---

3. VISUAL SIMILARITY

In this section we present a novel visual similarity measure that is one of the pillars which the instance learning approach proposed in this paper is based on. The main ideas which the visual similarity measure if founded on are: (i) a data record can be seen as a set of visually similar PNodes; (ii) two set of PNodes are visually similar if they contain nodes which leaves are spatially arranged in similar way on the rendered Web page and that have similar presentation features. Spatial arrangement of PNodes are computed by using the Rectangular Cardinal Relation model (RCR) [3] widely adopted in spatial reasoning. Roughly speaking, given a set of PNodes, the spatial context represents, for each pair of PNodes in the set, reciprocal RCRs. The spatial context of a set of PNodes is computed by means the function $\text{Context }: 2^V \rightarrow 2^V \times V \times V$, that for each pair of PNodes in the input set of PNodes compare coordinates and computes direct and inverse RCR relations.

4. THE SILA ALGORITHM

In this section we present the spatial instance learning algorithm that extracts data records and items from Deep Web pages by exploiting visual patterns created by Web designers in order to help human readers in making sense of Deep Web pages contents. The Algorithm 1 takes as input a PDOM and returns a set of data records instances with aligned data items.

Algorithm 1. InstanceLearner

Input: A PDOM $P$; Output: A set $I$ of data records instances with aligned data items.

1.1: $R_s := \text{findDataRegions}(P, \lambda)$;
1.2: $R := \text{maxRegion}(R_s, \mu)$;
1.3: $R_s = \text{similarRegions}(R, R_s - R)$;
1.4: for all $(R \in R_s)$ do $rs := rs \cup R.records$;
1.5: $I := \text{getDataItems}(rs)$;
1.6: return $I$.

Algorithm 1 consists of the two steps described below:

1. Data region and data record identification. In this step (instructions 1.1-1.3), the PDOM of a Deep Web page is taken as input and a set of data regions, which are portions of Deep Web pages containing similar data records, are returned. The procedure findDataRegions collects PNodes that represent data regions by performing a depth-first search in the PDOM in input. The procedure maxRegion takes as input found data regions and a threshold $\mu$, that represents the minimum number of records admitted for a data region, and chooses the region $R$ that has the greatest area. Such a choice is due to the fact that in Deep Web pages the size of the most important data region is larger than the size of the areas of other possible data regions. Since, smaller data regions can contain the same type of records that are in the main region, the method similarRegions finds smaller regions similar to $R$. This way the instance learning algorithm allows for finding data records spread in multiple regions and discards only not relevant information.

2. Data records and data item extraction. In this step (instructions 1.4-1.6), the algorithm constructs final data records in which data items of the same semantics are aligned by means of the procedure getDataItems.

5. EXPERIMENTS AND DISCUSSION

The instance learning approach presented in the paper has been experimentally evaluated on a dataset of 210 Deep Web Pages randomly selected from most known Deep Web Sites. Table 5 shows results obtained by applying the SILA algorithm, and compares such results with those obtained on the same dataset by applying MDR [1] and ViNTS [7] approaches. It is noteworthy that versions of MDR and ViNTS available on the Web allow for performing only data record extraction. Precision and Recall of SILA, in extracting data records, have been 99.79%, and 98.19% respectively. Whereas, MBR has Precision 25.61%, and Recall 46.75%, and ViNTS has Precision 53.57%, and Recall 50.47%. Precision and Recall of SILA, in extracting data items, have been 95.68%, and 99.40% respectively. Whereas MDR and ViNTS are unable to extract data items.

<table>
<thead>
<tr>
<th>Sites</th>
<th>Data Records Extraction</th>
<th>Data Items Extraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>SILA</td>
<td>P</td>
<td>R</td>
</tr>
<tr>
<td>MDR</td>
<td>99.79</td>
<td>98.19</td>
</tr>
</tbody>
</table>

Table 1: Precision and Recall Scores for 210 pages from 70 Deep Web sites

6. CONCLUSIONS AND FUTURE WORK

In this paper has been presented SILA a novel spatial instance learning approach for Deep Web pages. SILA is based on: (i) the novel Positional Document Object Model (PDOM) that represents both spatial and presentation features of data records and data items/fields produced by layout engines of Web browser in rendering Deep Web pages; (ii) the novel visual similarity measure that computes similarity between sets of PDOM nodes by exploiting their spatial and presentation features. Such a measure considers two set of PDOM nodes similar if leaf nodes they contain have a similar spatial arrangement and similar presentation features on the Deep Web page. This visual similarity measure exploits the rectangular cardinal relation model [3] widely adopted in spatial reasoning. So at the best of our knowledge this is the first paper in which models and results coming from the spatial reasoning fields have been applied to automatic instance learning on the Web and Web information extraction. Experiments carried out on 210 Deep Web pages randomly selected from well known Deep Web sites, show very high precision and recall.

The adopted dataset, and a more complete version of experimental results, are available at www.icar.cnr.it/ruffolo/SILA/dataset
7. REFERENCES


