Data Integration in the Wild: From Instances to Concept Catalogs

Patrick Pantel, Andrew Philpot and Eduard Hovy
USC Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292
{pantel,philpot,hovy}@isi.edu

ABSTRACT
Internet users today have access to seemingly unlimited amounts of information, and yet, due to the lack of companion structure, such information often cannot be effectively located or used. In response, we propose and demonstrate one component of a framework for autonomously learning and classifying instance data drawn from document collections on the Web. We have already applied this general-purpose prototype to extract conceptual representations from various websites for DVD products, hotels, restaurants, people, and cars.

Categories and Subject Descriptors
I.2.7.i [**Natural Language Processing/Web text analysis**].

General Terms
Algorithms, Experimentation.

Keywords
Data integration, concept learning, web mining.

1. DATA INTEGRATION IN THE WILD
We live in a digital society where information is streaming around us at an unprecedented scale forever changing the way we interact with the world. But much of the information on the Internet, especially data in textual form, is not effectively accessible without laborious and error-prone effort. In this new age, the old problem of data integration has re-emerged as one of the core problems in computer science. This paper presents an ambitious model for general-purpose data integration *in the wild* where computer programs crawl pages on the Web, automatically build conceptual models for the pages, and then integrate the information across several pages into unified concept catalogs.

In an attempt to address the inherent lack of structure and organization of web data, several technologies have been developed but often fail to both provide the necessary breadth of coverage of a search engine and the coherent integrated data views of a domain specific integrator (e.g., Nextag).

Digital government is by no way immune from this phenomenon. Users of government data have great difficulty navigating, organizing, and correlating the vast quantity of information that the government maintains. A large proportion of the Internet data of interest to both DG and general users is ground level *instance* data. A user looking for reviews of a movie, the price of jet fuel, the list of comments regarding a proposed government regulation, the forms required by various jurisdictions to incorporate a small business, etc., is essentially searching with a partially instantiated schema, for which she hopes to find the full specification.

In response, we propose an ambitious Web-scale framework to *a priori* autonomously learn, create, classify and index the sets of ground level instances that users are likely to be interested in by learning conceptual representations of Web pages. In this demonstration, we will showcase a prototype implementation of a key component of the architecture: the learning of conceptual models from a body of instances and the extraction of instances based on these models.

2. BUILDING CONCEPT CATALOGS
Figure 1 illustrates our proposed framework. This project demonstration showcases a prototype implementation of Phase 2. Phases 1 and 3 are conceptual. In Phase 1 of the framework, autonomous agents crawl the Web and cluster pages within subdomains. For example, given a subdomain such as Amazon.com, the agents will cluster all pages that it crawls from the site. The goal here is to group together all pages within a subdomain that are conceptually similar (e.g., all DVD products on Amazon.com would be a desired cluster). Other examples include extracting a list of staff members in a given organization or all the facilities which report emissions to the EPA. Although the framework allows for this process to take place on all pages on the Web, it is possible to focus the crawling on particular sites of interest (e.g., websites that collect opinions on public policy).

The goal of Phase 2 is to learn conceptual representations for the clusters output in Phase 1. By manually seeding this module with collections of web pages of a given class (e.g., DVD products from Amazon.com), our prototype implementation automatically learns a conceptual representation for the class (e.g., DVD's have a *release date*, *price*, *actors*, *director*, etc.) and then extracts for any page of this class instances of the conceptual representation (e.g., the DVD *Cars* was released on *November 7, 2006*).

Figure 2 shows a sample data flow and output of our system for the Amazon.com DVD example described above. Our system starts by learning a conceptual representation of the set of pages by identifying the semantic types of terms (e.g., *named entities*, *prices*, *dates*, *zip codes*, …) and then proceeds to identify anchor points, which are terms or semantic types that are found in most of the input pages. If an anchor point is followed by varying strings (or *fillers*)[1] of a similar identified semantic type across the pages, then the anchor point is identified as a semantic property for the conceptual representation of the pages. The expected semantic type of the filler is also extracted with the anchor.

[1] If the filler of an anchor does not vary, then the system rejects the anchor as a semantic property, helping avoid extracting items such as footers and breadcrumbs as semantic relations.

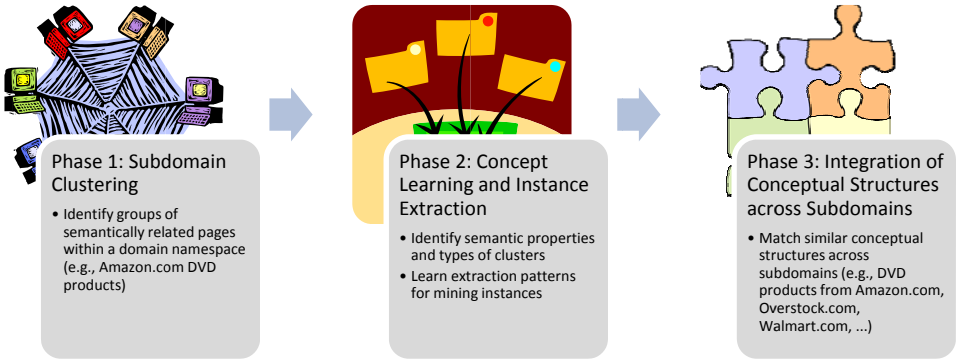

Figure 1. A model for data integration in the wild. Collections of pages (e.g., the Web) are clustered, conceptual representations are learned, instances are extracted, and data is integrated across the collection using the conceptual models.

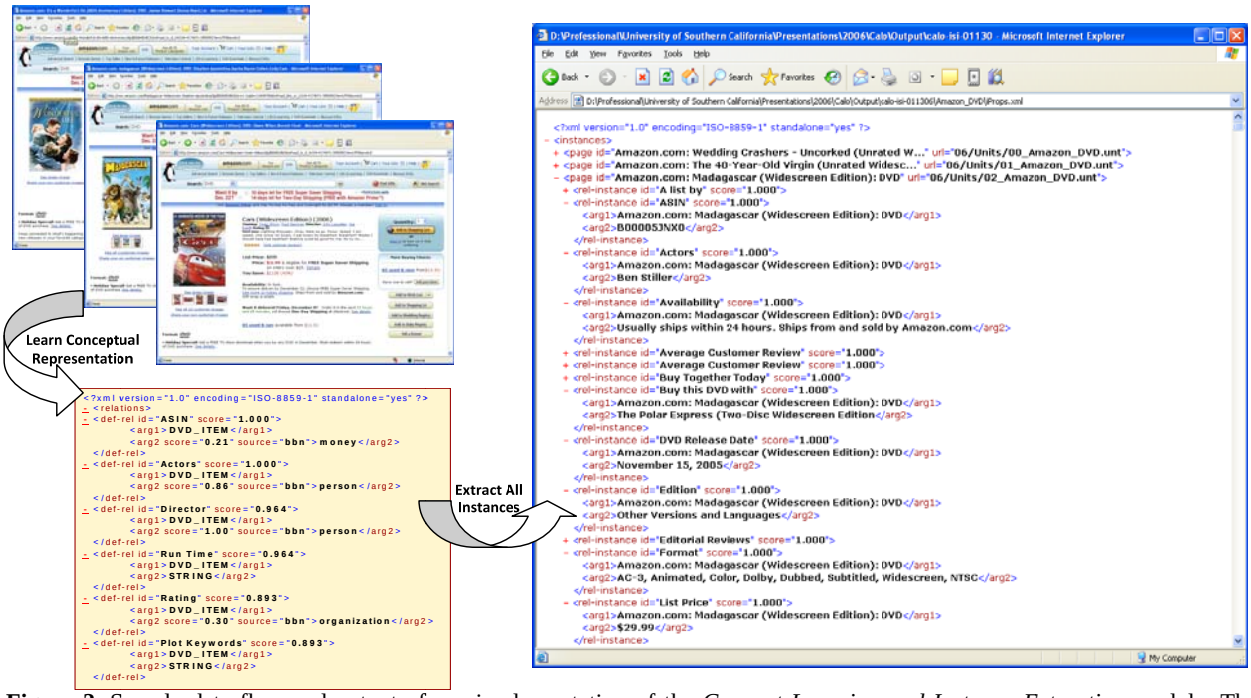

Figure 2. Sample data flow and output of our implementation of the *Concept Learning and Instance Extraction* module. The example is drawn from manually extracted Amazon.com DVD products.

For example, since most DVD product pages on Amazon.com have an anchor *Release Date* which is followed by strings identified as *dates*, and since the dates vary between the pages, the system is able to extract that the input pages have a property *Release Date* which takes a *date* as its filler. Given these semantic properties, the system can also extract regular expressions (surface patterns) to locate the semantic property's filler in any new page[2]. These regular expressions are then used to extract instances of the semantic properties in all given pages as well as new pages (e.g., new DVD's released on Amazon.com). To date, our system has been able to extract conceptual representations and extract instances for the following domains: cars on ebay.com, hotels on frommers.com, people at SRI, and restaurants on CitySearch. In this demonstration, we will showcase this system on a set of subdomains of interest to the DG community.

Finally, Phase 3 matches conceptual representations of clustered pages to build unified concept catalogs cross-cutting the various sources of data on the Web. For example, this Phase would unify the representations of DVD products from all sites such as Amazon.com, Overstock.com and Walmart.com.

Our long-term dream is to enable wide-open integration of information in a wide range of contexts behind-the-scenes, providing simplified access to information for the masses.

[2] These patterns are unfortunately susceptible to failure due to format or label changes on new pages. To handle this, we envision a feedback module capable of detecting when new pages no longer express a semantic property and then re-running the conceptual representation learning module.