Phoebus: A System for Extracting and Integrating Data from Unstructured and Ungrammatical Sources*

Matthew Michelson and **Craig A. Knoblock**
University of Southern California
Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292 USA
{michelso,knoblock}@isi.edu

Abstract

With the proliferation of online classifieds and auctions comes a new need to meaningfully search and organize the items for sale. However, since the seller's item descriptions are not structured and do not conform to a standard set of values (think "Chevy" versus "Chevrolet"), searching and organizing this data is difficult. This paper describes a working demonstration of the Phoebus system which uses both record linkage and information extraction to parse out the meaningful attributes of an item description and assign them standard values. This allows the data to be sorted, searched and linked to other data sources where standard values for the attributes are required to link the sources together.

Introduction

The huge popularity of online classifieds and auctions has proved to be a tremendous benefit to sellers who can reach ever larger markets for their goods and this industry is booming. For example, the classifieds list Craig's List[1] has listings all over the world, and the auction house EBay[2] is a major corporation. However, from a buyer's perspective, while there are more products to choose from, finding a particular item can be difficult. The sellers often describe their items without regard to structure or standard values for the items they are describing. For example, consider the three listings for different Honda Civics, shown in Table 1, that are taken from Craig's List.

Craig's List Post
93 civic 5speed runs great obo (ri) $1800
93- 4dr Honda Civc LX Stick Shift $1800
94 DEL SOL Si Vtec (Glendale) $3000

Table 1: Three posts for Honda Civics from Craig's List

Clearly, searching posts by keyword is inadequate. For one, the keywords might be misspelled, as with "Civc" in the second post. Also, standard values are not used for some of the important attributes. For example, the "DEL SOL" of the third post actually refers to a Civic Del Sol, so if a user searches for Civic, this car should be returned as well. Lastly, the first and third posts are actually missing the make, Honda. So a buyer searching with the keyword Honda will miss both of these listings, even though they are relevant.

Furthermore, without standardized attributes, other data sources can not be linked into the listings. For example, we might like to link a post to an external website that lists useful information about the item for sale. This external source might be queried by a URL with the attributes as parameters embedded within it. Without the proper standardization, however, this external source could not be integrated.

This paper presents a demonstration of the Phoebus system(Michelson & Knoblock 2005), which "semantically annotates" unstructured sources. Semantic annotation adds attributes to a data source by extracting and tagging the attributes of interest from that source. To do this, Phoebus exploits outside collections of entities, called "reference sets." Using our car example, our reference set could be the set of cars provided by a trusted authority, such as the Edmunds car buying guide.[3]

To exploit a reference set, Phoebus employs a two step approach. First it matches the post to the best matching member of the reference set. This is known as record linkage. Once a post is matched to the member of the reference set, it has standard values for the attributes it is matched on. Furthermore, the matching record from the reference set can provide attributes that were not included in the post. As with the first post of Table 1, the make was not included, but by matching it to a correct member of the reference set, it now

*This research is based upon work supported in part by the National Science Foundation under Award No. IIS- 0324955, in part by the Defense Advanced Research Projects Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under Contract No. NBCHD030010, in part by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-00-1-0504, and in part by the Air Force Office of Scientific Research under grant number FA9550-04-1-0105. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any of the above organizations or any person connected with them.

[1] www.craigslist.org

[2] www.ebay.com

[3] www.edmunds.com

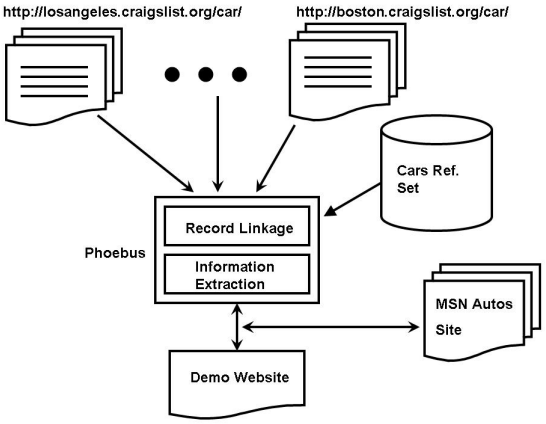

Figure 1: The architecture of the demonstration

Figure 2: A screen shot of the Phoebus demo website

has a make and can be searched on that attribute.

After record linkage, we perform information extraction to identify and parse out both the attributes from the reference set and attributes not easily represented in a reference set, such as prices and dates. By extracting the attributes themselves (instead of stopping at record linkage) we can see the actual included values and we more accurately extract the attributes not in the reference set. For example, if a car was called the "$50" as a model, the system would know that this is actually a model, and not a price. For more details please see (Michelson & Knoblock 2005).

Once Phoebus annotates a data source we can query the source and integrate it with outside sources. The standard set of attribute values allows Phoebus to treat the unstructured source as a database and allows outside sources to be integrated using these standardized values.

Semantic annotation of web sources is a studied problem (Cimiano, Handschuh, & Staab 2004; Vargas-Vera *et al.*), but most systems rely on natural language processing to parse out the attributes. With our unstructured data this is not an option because users do not generally include any common structure or grammar in their posts.

This work is also similar to data cleaning e.g.(Chaudhuri *et al.* 2003). However, data cleaning approaches rely on a function that maps the attributes of one record to the attributes of another. Since our data is unstructured, and the attributes are embedded in the post, these approaches will not work in our case.

Querying and Integrating Sources

To demonstrate Phoebus, we built an example website[4] that allows users to search, sort and query posts from Craig's List about used cars for sale in six cities. We also link the results to an outside source, MSN's Auto section, which provides further information about the car, such as it's quality rating. Figure 1 shows the demo's architecture and Figure 2 shows a screen shot of the results page.

[4]http://www.isi.edu/integration/Phoebus/demos.html

Query support

With Phoebus, users can search Craig's List by make, city and price, and then sort the results by post, make, model, trim, year, price, and date posted. This searching and sorting demonstrates querying the original posts by any of the attributes embedded within it. So, even if the posts contain misspellings or missing attributes, we can still correctly search them. In fact, we could even support aggregate queries such as "How many Honda Civics are under $5000?"

Integration

This demonstration shows two types of integration. The first is meta-level searching. Since each source used in the demonstration provides it's own set of posts to Phoebus via RSS, which are then searched all at once, it is easy to imagine bringing in other sources, such as EBay Autos, to create a meta-search over car listing sources.

Second, we integrate the MSN Auto site to show how to integrate external sources that are linked with common attributes and values. While we link to the external page itself, it is easy to imagine writing a wrapper that scrapes the information in real time, integrating it into our car search. It is key to note that the URLs for MSN Auto are constructed at query time based on the values in the post. This shows that we actually integrate MSN Auto rather than just pulling up a URL that we constructed offline.

References

Chaudhuri, S.; Ganjam, K.; Ganti, V.; and Motwani, R. 2003. Robust and efficient fuzzy match for online data cleaning. In *Proceedings of ACM SIGMOD*.

Cimiano, P.; Handschuh, S.; and Staab, S. 2004. Towards the self-annotating web. In *Proceedings of WWW-2004*.

Michelson, M., and Knoblock, C. A. 2005. Semantic annotation of unstructured and ungrammatical text. In *Proceedings of IJCAI-2005*.

Vargas-Vera, M.; Motta, E.; Domingue, J.; Lanzoni, M.; Stutt, A.; and Ciravegna, F. Mnm: Ontology driven semi-automatic and automatic support for semantic markup. In *Proceedings of EKAW-2002*.